MICROHOBBY

Ano I — Nº 11 — 1984
Cr$ 1.800,00

DRAGÃO

É tão Bonzinho... O Dragão

Para Manaus, Boa Vista, Porto Velho, Rio Branco, Santaré, Altamira, Macapá, Est. de Rondônia (Via Aérea), Cr$ 2.340,00

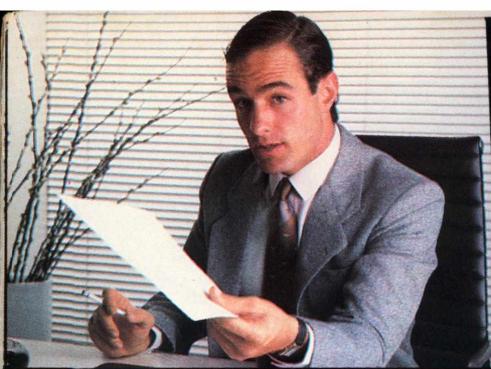
Números Complexos: sem Complexos

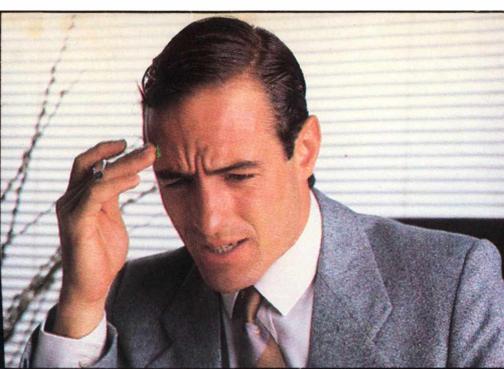
Multiestoque – para o TK-2000

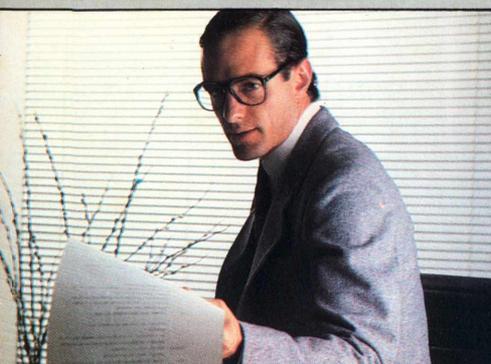
Amor do Sr. Nabor: Quem será?

Dicas: Apagando Linhas em Bloco

Reversão: O que existe no Espelho?

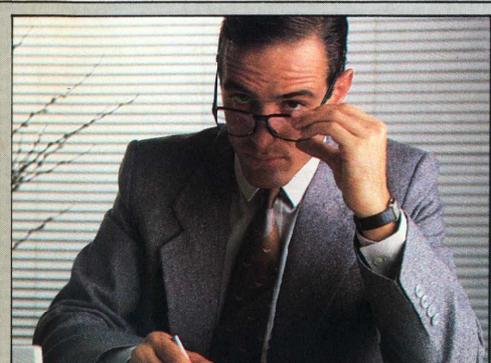
Reversão: O que existe no Espelho?

"Bata esta carta para a diretoria. Com todos os acentos e sem erros!!!"

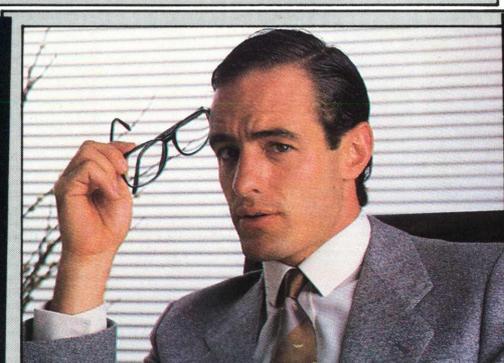
BZZZZZ.
BZZZZZ.
PRONTO!

"Meu Deus! Preciso de mais 15 cópias deste relatório, e o presidente vai começar a reunião daqui a 5 minutos!"

BZZZZZ.
BZZZZZ.
PRONTO!

"Quero outro tipo de letra. Esta carta, além de importante, é muito pessoal."

BZZZZZ.
BZZZZZ.
PRONTO!

"Faça estes gráficos para mim. Daqui a meia hora tenho reunião de vendas."

BZZZZZ.
BZZZZZ.
PRONTO!

"Aquelas tabelas têm que ser refeitas. A diretoria mudou todas as previsões. Por favor, rápido!"

BZZZZZ.
BZZZZZ.
PRONTO!

"Lembra daquela circular que você bateu o mês passado? Preciso de mais 5 cópias dela para as filiais, já com nossos novos endereços!"

BZZZZZ.
BZZZZZ.
PRONTO!

Se você tem ou pretende adquirir um micro para o seu escritório, não deixe de conhecer a Mônica da Elebra Informática. Melhor do que qualquer máquina de escrever ela passa para o papel todo o talento do seu computador, e você ainda ganha uma secretária mais feliz de presente.

PARTNER

Mônica.* A secretária da secretária eficiente.

Mônica. Impressora serial da Elebra Informática. 100 caracteres por segundo/Impressão bidirecional, com procura lógica/Caracteres comprimidos e expandidos/Capacidade Gráfica/Qualidade Carta/Livre escolha de tipos pelo usuário/Compatível com a maioria dos melhores microcomputadores nacionais. Imprime em português correto, com acentos, cedilhas e tudo.

**com Tipograf. Opcional na EI 6010.*

Filiada a ABICOMP

elebra informática

Elebra Informática S.A. - Av. Engº Luís Carlos Berrini, 1461 - Tels.: (011) 531-4844/4889/4975 - CEP 04571 - Telex: (011) 25957 ELIN - BR - São Paulo - SP.
Av. Rio Branco, 50 - 10º andar - Telefone: (021) 253-5596 - CEP 20.090 - Telex (021) 23481 DOCA - BR - Rio de Janeiro - RJ.

EDITORIAL

Há mais de um ano me vi incumbido de estruturar uma revista dedicada a usuários de microcomputadores pessoais.

Como professor (e não jornalista!) senti a necessidade de *formação* e não *informação* destes leitores. Para informação já existia uma série de boas revistas no mercado, desde a tradicional e excelente MICRO-SISTEMAS, até a inteligente BITS, passando pela MICRO-MUNDO, INFO e tantas outras.

Modelos estrangeiros foram consultados e abandonados: tivemos a coragem de confiar em nossa própria sensibilidade e, ao conceber MICROHOBBY, não copiamos ninguém! Obviamente aproveitamos idéias e programas publicados em outras revistas (aliás é para isto que eles são publicados, ou seja, dados ao público) sempre que possível citando a fonte e sempre adaptando-os e melhorando-os, às vezes tanto que os tornamos irreconhecíveis.

Ao publicarmos livros, demos ênfase ao autor nacional, evitando a solução editorialmente pobre da tradução fácil e nem sempre bem feita.

Passamos, obviamente, por uma série de dificuldades características de uma empresa nova e em rápido crescimento: atrasos, problemas em fitas-brinde, problemas com o correio, etc. Mas afinal, se a "Lei de Murphy" é Universal, por que não haveria de se aplicar a nós também?

Na parte editorial, porém, sob a minha direta responsabilidade, nunca medi esforços para que a qualidade das matérias fosse sempre do mais alto nível e de máximo aproveitamento para nossos leitores.

Com relação a isso devo admitir que fui amplamente recompensado: todas as cartas, repito, *todas* as cartas por nós recebidas, mesmo as enviadas para veicular algum tipo de reclamação, sempre nos transmitiram elogios e estímulo.

Por isso, ao deixar a direção editorial da MICROHOBBY, pressionado entre outras coisas por uma série de afazeres que transformaram minha vida numa crônica "falta de tempo", saio contente e com a sensação de ter feito, entre os milhares de assinantes, milhares de amigos.

Vou me dedicar mais ao Núcleo de Orientação de Estudos, onde uma série de projetos em desenvolvimento reclamam minha presença: a parte didática da informática foi, até agora, muito pouco desenvolvida e está a reclamar soluções e idéias originais e inteligentes.

Deixo a Revista e a Editora em boas mãos, encerrando este último editorial com uma frase que pode ser considerada estereotipada mas expressa o que sinto:

VALEU A PENA!

Pierluigi

Microdigital e novidades para os usuários dos TKs

Com o TK-2000 à frente, a Microdigital está lançando no mercado, inúmeras novidades para os usuários não só deste micro mas também para os dos outros TKs. A empresa está colocando no mercado a expansão de 48 K para o TK-85; o *Joystick* com movimentação nas 4 direções, que permite um maior dinamismo nos jogos; a *interface paralela* que permite a ligação dos micros com as impressoras do padrão Centronics, e o *gravador de EPROM,* uma interface que, acoplada ao conector de expansão permite ao usuário a conversão do TK 83 e 85 em uma ferramenta de trabalho.

Além destas novidades, a Microdigital está enviando o manual técnico do TK-2000 a todos os usuários que compraram o micro e enviaram o cartão de garantia para a empresa. Este manual, segundo os diretores da empresa possui inúmeras dicas de funcionamento do TK-2000; sobre o seu software, suas rotinas, sobre alta-resolução e sobre o dissassembler, além de inúmeras outras informações que orientam o usuário da melhor forma possível. Os computadores que estão saindo da fábrica, já trazem o manual incorporado a eles.

O Apple atuando como terminal de um IBM

A COPEC — Computadores, Programas e Comércio S.A. está lançando no mercado um pacote de hardware e software que transforma o Apple e seus compatíveis em um terminal inteligente para computadores IBM.

O Software para o TK-2000

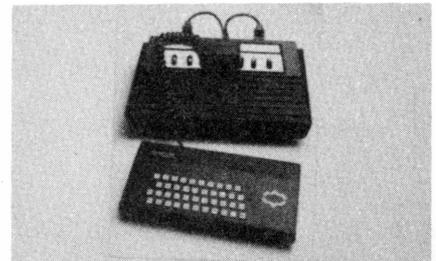
Dentre as empresas que estão oferecendo produtos para serem utilizados no TK-2000, a MultiSoft talvez seja a que tem a maior quantidade de software disponível. A empresa está fornecendo aos usuários do TK-2000 inúmeras fitas (com nova embalagem) que vêm com jogos, utilitários e aplicativos. Do software já disponível, a empresa tem entre jogos: *Sabotagem; Ataque; Auto-Estrada; Bombardeio; Corrida; Eliminator; Fliperama; Multi-Invader; Pânico; Papa-Tudo;* e *Pulo do Sapo.*

De utilitários e aplicativos, a empresa conta com programas como: *Multicad; MultiEstoque; Editor* e o *Software-impressora,* um programa destinado a acionar o hardware de impressoras no TK-2000.

Dentro do software que estará disponível para o usuário em greve, a MultiSoft irá oferecer programas de jogos como: *Xadrez; Gamão* e *Poker; Estação Espacial Orbital* entre outros. Dentre os aplicativos e utilitários destacam-se *Contas a Pagar, Mala-Direta* e outros.

O videogame virando microcomputador

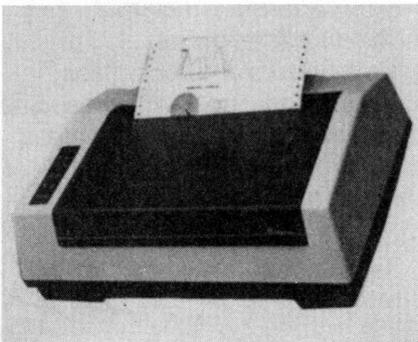
A Milmar que apresentou em março passado, o seu microcomputador Apple-Senior, dotado de teclado auxiliar e compatível com os sistemas CP/M e Applesoft, lançou uma novidade bastante interessante, principalmente para quem possui videogame: o *Dactar Comp.*

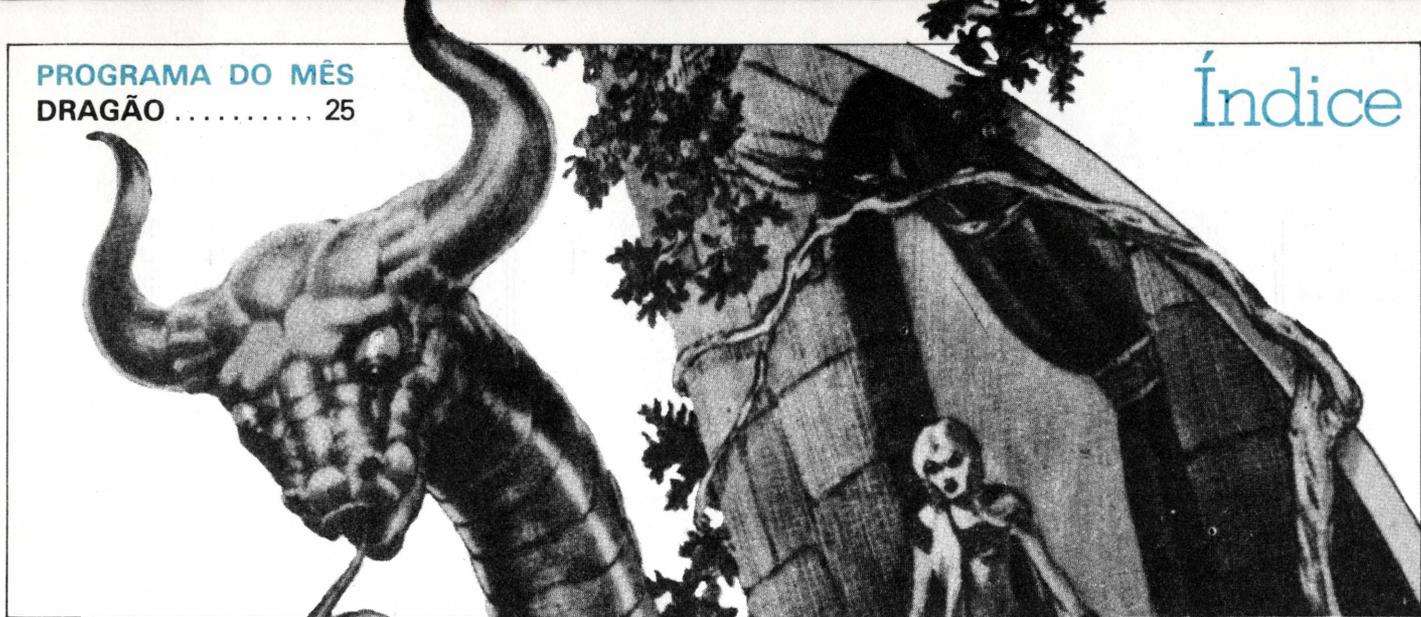
O *Dactar-Comp* pode ser considerado uma interface de videogame, ou seja, ele é "micro" com teclado de 42 teclas com funções pré-programadas que, adaptado ao videogame complementa-o em mais de 2 K de RAM e 16 K de ROM. Assim, a interface pode compor músicas de até 8 oitavas em 2 canais, fazer figuras e gráficos com capacidade gráfica de até 10 cores diferentes.

A Andei e o Software/84

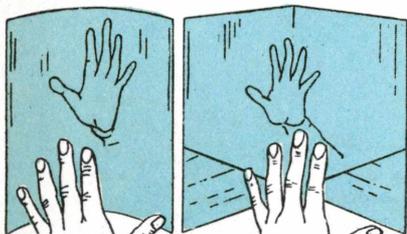
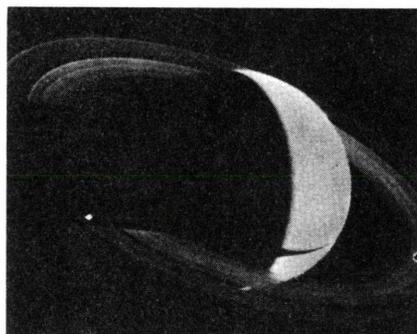
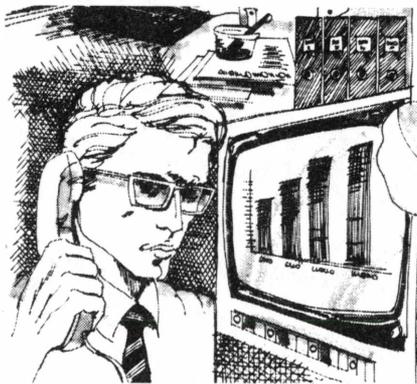
A ANDEI — Associação Nacional dos Dirigentes e Executivos de Informática está instituindo um prêmio denominado "O Melhor Software/84". O prêmio é patrocinado pela Itautec e tem como objetivo a divulgação dos trabalhos projetados e implantados, sejam programas ou conjunto de programas.

Os softwares devem permitir utilização em mini computadores, ter memória de no máximo 64 kB, disquetes de 8' ou 5 1/4' e impressoras com até 132 posições.

Para concorrer ao prêmio, a entidade informa que o candidato deve enviar até o dia 30/9 para a Rua Capitão Antonio Rosa, 376 — conj. 102, os seguintes documentos: listagem dos programas, relatórios ou documentos emitidos e preencherem formulário de inscrição fornecido pela ANDEI.

A Mônica e a Campanha da Reserva

A Elebra, fabricante das impresoras *Mônica* e *Emília* lançou em maio último, o logotipo-base de toda sua campanha institucional: uma bandeira nacional estilizada. A bandeira transmite a proposta da empresa que é a de defender e valorizar a indústria nacional de informática.

A empresa estará presente no Micro-Festival/84 do Rio de Janeiro, apresentando várias novidades. Entre estas, a impressora *Mônica* com o módulo opcional *Tipograf,* um lançamento que enriquece a impressora e permite, por exemplo, a mudança do tipo de caractere durante a impressão com apenas um comando: direto do micro para a impressora. Além do módulo a Elebra que detém 60% do mercado de impressoras no Brasil, apresentará a outra Mônica com carro mais longo (15 polegadas) e a *Unidade* de *Disco Flexível* 5,25" — *Horácio* — para utilização em micros da linha Apple e TRS.

Índice

Capa Cassiano Roda

Expediente

DIRETOR EDITORIAL
Pierfuigi Piazzi

JORNALISTA RESPONSÁVEL
Ana Lúcia de Alcântara — Mt. 14495

EDITOR
Álvaro A. L. Domingues

ASSESSORIA TÉCNICA
Flávio Rossini
Wilson José Tucci

ANÁLISE E REDAÇÃO
Nancy Mitie Ariga
Carlos Eduardo Rocha Salvato
Gustavo Egídio de Almeida
Renato da Silva Oliveira
Roberto Bertini Renzetti

PRODUÇÃO GRÁFICA
José Carlos Sarkis

ARTE
Cassiano Roda
Eliana S. Queiroz Yoshihara
Fátima M. Rossini Gouveia
Osmère Sarkis

CORRESPONDENTES
New York — Flávio Rossini

PUBLICIDADE
Aurio J. Mosolino
Eduardo Garcia de Souza

CIRCULAÇÃO E ASSINATURAS
Marcia Regina Dominiqui

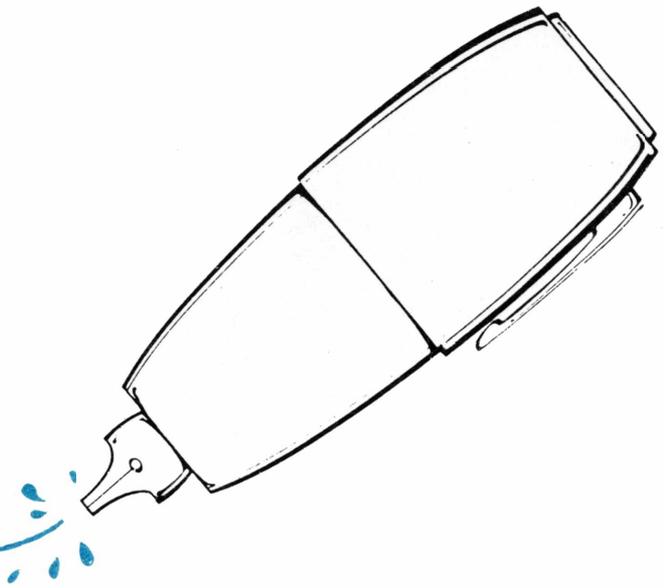
GERÊNCIA GERAL
Angel Zaccaro Conesa

DISTRIBUIÇÃO
Fernando Chinaglia Distribuidora

MICROHOBBY é editada mensalmente por MICROMEGA PUBLICAÇÕES E MATERIAL DIDÁTICO LTDA., INPI 2992 Livro A
Endereço para correspondência:
Rua Bahia, 1049 — Cx. Postal 54096 — CEP 01296 — São Paulo, SP.

Para solicitar assinaturas (12 números) envie cheque nominal à MICROMEGA P.M.D. LTDA., no valor de Cr$ 19.800,00.

NÚMERO 11

CARTAS DOS LEITORES

TK 2000

Gostaria de saber se o TK-2000 é totalmente compatível com o Apple e seus compatíveis nacionais e, se sim, porque tamanha diferença de preço. Além disso, quero saber suas opiniões no que toca esta e outras diferenças entre computadores.

Sugiro ainda a análise, na seção Fita do Mês, de um programa compilador, qual seu princípio básico, etc. . .

Ricardo Castro — Belém, PA.

Caro Ricardo

*Tudo o que você quer saber sobre o Apple e o TK-2000 foi discutido no artigo **TK-2000: um compatível com o Apple ou uma nova Máquina?**, publicado no nosso número 8. Mais informações sobre o TK-2000 estarão, agora regularmente, nas páginas de Microhobby.*

Já publicamos, na Microhobby número 7, a análise de um programa compilador comercial, na seção Fita do Mês.

Observação:
A aula 11 do Curso de BASIC sairá na revista 12, dando continuidade às lições.

CARTA DO EDITOR

Informamos que este é o último número em que o Sr. Pierluigi Piazzi comparece no nosso expediente como Diretor Editorial. Obrigações outras o impedirão de continuar com a Microhobby, mas isso não o afasta completamente de nós já que continuamos mantendo contato com ele.

A partir deste número, assumo plenamente as funções de Editor da revista, função que vinha exercendo sob a supervisão do Sr. Pierluigi desde o número 6 de Microhobby.

Acredito que existem basicamente dois compromissos que devo assumir ao dar continuidade ao andamento da revista Microhobby. O primeiro deles refere-se à manutenção do espírito da revista. A Microhobby é uma revista de caráter didático, que preocupa-se em introduzir seus leitores no mundo da Informática, tratando com seriedade assuntos sérios e com humor onde for possível.

O segundo compromisso são as mudanças que são necessárias e que serão implementadas. Isto não é de se estranhar, visto que a mudança para melhor é uma característica da Microhobby desde seus primeiros números.

Assim, conto com a sua colaboração para que eu possa assumir estes dois compromissos, sem perder de vista as necessidades dos leitores, para daqui a um ano dizer, como o Sr. Pierluigi Piazzi, que valeu a pena. Só espero que, quando chegar esta ocasião eu não esteja escrevendo um editorial de despedida, mas o primeiro de uma nova fase, neste contínuo evoluir que é a Microhobby.

O Editor

Outros computadores

Evoluir é conhecer. Como é possível evoluir na ciência da computação conhecendo (mesmo bem) somente o TK 85? Lembre-se que, no mercado brasileiro, já temos quase uma centena de microcomputadores.

A intenção do leitor Leonardo C. Linhares *(Cartas dos Leitores,* Microhobby 8) em "reunir e aproximar" é muito louvável, porém creio que esta tarefa compete aos clubes e não à uma revista.

Roberto Massaru Watanabe — São Paulo, SP.

Caro Roberto,

Agradecemos a você a resposta à carta do leitor Leonardo C. Linhares.

A Microhobby nunca procurou fugir a dois propósitos: dar um apoio substancial aos possuidores de TK e fornecer-lhes subsídios sobre computação. Dê uma olhada na seção Vice-Versa do número anterior, onde falamos de como o TK pode ser aproximado do TRS-80 mediante o conhecimento das duas máquinas.

Anúncios

Como editar desta revista, apesar do preço não justificado pelo enorme número de anúncios, gostaria de parabenizá-los pelo ótimo nível da revista, esperando e torcendo por um maior número de programas em Linguagem de Máquina e dicas, os quais julgo de extrema importância para todos os iniciantes nesta linguagem.

Caro Fernando,

Agradecemos os elogios e críticas à nossa redação. Entretanto, não concordamos que o preço da Microhobby seja injustificado. Afinal, o que você recebe em nossas páginas é sem dúvida muito mais valioso do que uma fita contendo vários programas. Além disso, não achamos o número de anúncios excessivo.

Quanto às dicas e aos programas em Linguagem de Máquina cremos que temos publicado o suficiente para agradar a maioria dos nossos leitores. De qualquer forma, na medida do possível, eles aumentarão em número e qualidade.

Pirataria

Atualmente não existe uma legislação a respeito de programas de computador. Isso prejudica o programador, que vê seus programas sendo manipulado por pessoas estranhas e não pode tomar nenhuma medida contra. (. . .)

Sérgio S. Caixa
Rio de Janeiro, RJ

Caro Sérgio,

Realmente o assunto é controvertido, não sendo este problema exclusivamente brasileiro. Existe muitas dúvidas entre juristas do mundo inteiro em como classificar um programa: se como obra intelectual, protegível pelo direito autoral, ou como propriedade industrial ("invenção tecnológica" — sujeito a patentes).

Convém que você leia o artigo "Proteção Jurídica de um programa de Computador", do advogado Eduardo J. Vieira Manso, publicado nas revistas Microhobby 5 e 6, onde seu autor, especialista em Direito Autoral, discute o assunto.

DESGRILANDO

O Rally e a RAMTOP

Comprei o programa Rally da Multisoft e não entendi o funcionamento. Escrevi à esta revista anteriormente e minha carta foi enviada ao fabricante.

De acordo com a resposta da Multisoft, o programa Rally é carregado em duas etapas.

Sei, por experiência própria, quando é carregado um programa, o programa que estiver gravado anteriormente será apagado.

Gostaria que a revista esclarecesse, não só a mim, o que acontece neste programa.

(. . . . .)

Walter Gomes dos Santos
Rio de Janeiro, RJ

Caro Walter,

O programa Rally é composto por duas etapas. Na primeira etapa é carregado um programa que altera o valor de uma variável do sistema, a RAMTOP. Esta variável "conta" ao computador aonde termina a memória. A primeira etapa do Rally modifica esta variável "enganando" o computador, reservando uma área da memória que não é afetada pelo comando NEW. Nesta área, o primeiro programa se auto-transfere. Uma vez feito isso, este programa carrega o próximo programa da fita automaticamente, desde que o gravador esteja ligado, na posição PLAY.

Alta resolução no TK-85

Desejo saber se há possibilidade de se colocar alta resolução no TK-85 por software ou outro modo, mas sendo acoplado no interior do computador e não externamente.

Agenor Prates Ferreira Neto
Piracicaba, SP

Caro Agenor,

Na área de hardware, sabemos que existe uma possibilidade para o TK-83, mas que envolve uma alteração drástica na distribuição e endereçamento da memória RAM interna do micro.

CP-200 e as expansões da Microdigital

Sou usuário do CP-200 e sei que este computador é compatível com o TK-85 e outros computadores que utilizam a mesma lógica. Tendo isso em mente, pergunto:

1) O que devo fazer para usar, no CP-200, a impressora do TK?

2) Existe meios de fazer com que o CP-200 ligue e desligue o gravador automaticamente?

3) É possível usar expansão de 64 kB do TK 83 no CP 200?

Edson Antonio Baldo
Nova Venecia, ES

Caro Edson,

Para ligar tanto a impressora como expansões do TK no CP-200 é necessário que se use um cabo apropriado, uma vez que os pinos da saída para expansões do CP-200 são diferentes do padrão utilizado no TK.

*Quanto a fazer com que o seu computador ligue e desligue o gravador é necessário uma interface especial ou modificações no seu **hardware**. Para fazer uma destas duas coisas, recorra a um técnico especializado. Alguns destes técnicos podem ter anunciado na seção Pequenos Anúncios. Dê uma olhada e verifique (cuidado com os "aventureiros"!).*

Economia de memória x tempo de processamento

Tenho lido em revistas do setor, várias formas de economia de memória, entre os quais se destaca o uso da instrução VAL, passando o programa a operar com variáveis *string* em lugar de variáveis numéricas.

Aplicando este princípio em um programa que elaborei houve um ganho enorme de bytes disponíveis quando substitui DIM A(1500) por DIM A$ (1500, 2) e, realmente passei a contar com uma área substancialmente maior para ampliar o programa. Entretanto, parece que houve um efeito colateral: o programa que dava o resultado em 30 segundos no modo FAST passou a dar o mesmo resultado em aproximadamente um minuto.

(. . .) Gostaria de um esclarecimento a este respeito e aproveito para sugerir um artigo sobre "como aumentar a rapidez de seus programas" (. . .)

Paulo Augusto M. Oncken
São Paulo, SP

Caro Paulo,

Normalmente, sempre que ganhamos alguma coisa em algum lugar perdemos outra em outro. É a lei da conservação da energia aplicada nas coisas práticas . . . No caso do TK, a maioria dos "macetes" usados para economizar memória correspondem à uma perda de tempo. Realmente existem "macetes" também para economizar tempo de processamento. Sua sugestão, neste sentido, é válida e vamos estudar o assunto para futuras publicações. Por enquanto, daremos algumas sugestões:

1) Trabalhe em FAST sempre que possível.

2) Uma vez que o programa esteja em sua forma final, elimine todas as linhas REM com comentários e numere as restantes de 1 em 1 (isso também economizará memória).

3) Evite parênteses desnecessários e mensagens de PRINTs muito extensas (novamente ocorrerá economia de memória).

*4) Trabalhe diretamente com instruções numéricas, sem recorrer a **strings** e funções VAL (aqui há perda de memória).*

5) Sempre que possível, use rotinas em Linguagem de Máquina.

Principiante

Como assinante desta revista quero elogiá-los, pois ela é ótima para um principiante em computação como eu.

Gostaria também que me respondessem o seguinte:

1) Quando for a época da renovação da assinatura vocês avisam?

2) Qual o tempo máximo que um TK-85 pode ficar ligado, sem prejudicar o seu funcionamento?

Wanderley Itamar Abrão
Orlandia, SP

Caro Wanderley,

Agradecemos seus elogios, acrescentando que a revista visa dar apoio significativo a todos os leitores que têm o computador como hobby, quer seja principiante quer não.

Pouco antes de sua assinatura terminar, você receberá um aviso, indicando quando deverá renová-la.

O TK-85 pode ficar muito tempo ligado. Soubemos de casos de pessoas que deixaram-no rodando um programa por mais de 12 horas, sem que houvesse problemas. Todavia, é conveniente "dar um tempo", desligando-o de quando em quando, caso não seja necessário rodar um programa muito extenso, ou no trabalho de digitação. Desligá-lo por 10 minutos à cada duas horas é um bom critério. Também é conveniente, na digitação de um programa muito extenso, gravar de quando em quando a parte já digitada. É terrivelmente frustrante perder, por qualquer motivo, o trabalho já digitado! ○

ESGOTO

Gustavo Egídio de Almeida

Memória ocupada — 1366 Bytes
Soma Sintática — 28233

Você caiu numa ***Tubulação de Esgoto***. Além de ser algo extremamente desagradável, você notará que, no meio das águas poluídas, virão ratos ao seu encontro. (Figura 1)

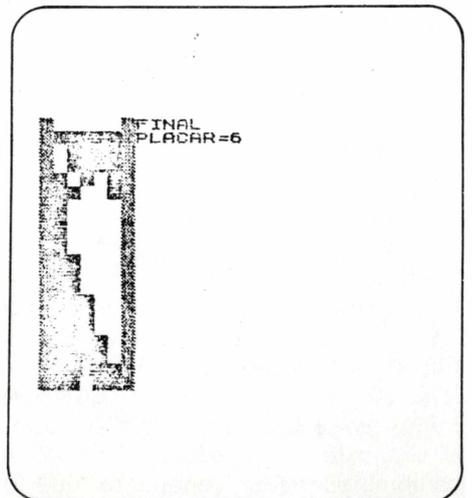
Você terá que desviar dos ratos para somar pontos, usando as teclas:

1 — Move o símbolo (▒) para esquerda

Ø — Move o símbolo (▒) para direita

Atenção aos gráficos usados:

Linhas: 25 — Gráfico H e espaço; 8Ø — Gráfico V inverso e * inverso; 1ØØ — Espaço inverso.

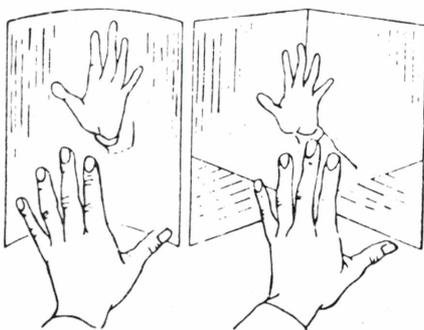
```
  5 LET DD=PI/PI
 10 LET S=PI-PI
 15 LET A=VAL "4"
 20 FOR B=VAL "1" TO VAL "20"
 25 PRINT " ▒        ▒"
 30 NEXT B
 35 LET D=INT (RND*VAL "5")+VAL
 "2"
 40 FOR C=VAL "19" TO PI/PI STE
P -DD
 50 LET E=INT (RND*3)+1
 60 IF E=2 AND D>2 THEN LET D=D
-1
 63 IF C<=VAL "5" THEN LET D=A
 65 LET A=A+(INKEY$="0" AND A<V
AL "6")-(INKEY$="1" AND A>VAL "2
")
 70 IF C=VAL "1" AND D=A THEN G
OTO VAL "200"
 80 PRINT AT VAL "1",A;"▒";AT C
,D;"▒"
100 PRINT AT VAL "1",A;"█";AT C
,D;"▒"
105 NEXT C
110 LET S=S+PI/PI
115 PRINT AT PI/PI,VAL "8";"PLA
CAR=";S
120 IF S>VAL "5" THEN LET DD=VA
L "2"
125 IF S>VAL "10" THEN LET DD=V
AL "3"
130 GOTO VAL "35"
200 PRINT AT PI-PI,VAL "8";"FIN
AL"
220 STOP
```

Esta Seção destina-se a fornecer subsídios para a quantidade já bastante significativa de professores secundários que usam computadores TK e compatíveis como recurso didático em seus cursos.

OÃSREVER REVERSÃO

Álvaro Csapo Talavera

Reversão na Natureza

As simetrias e as reversões são tão comuns na natureza que freqüentemente não as percebemos, tão profundamente envolvidos que estamos por elas. Por exemplo, nosso próprio corpo, ao menos na sua forma externa, se apresenta revertido em relação a um plano vertical, de tal forma que o nosso lado esquerdo é, muito aproximadamente, a imagem num espelho plano do nosso lado direito e vice-versa.

Este fato foi observado há muito tempo. Platão (427-347 A.C.) no "Timeu", Hierão (100-150 D.C.) em "Catóptrica", e Lucrécio (95-55A.C.) em "Sobre a Natureza das Coisas" já analisaram a reversão e chegaram a indicar a existência de espelhos onde ela deixava de ocorrer (por reflexão da reflexão), como os indicados na figura 1.

A importância da reversão para o ser humano ganhou um novo destaque com o desenvolvimento da fisiologia da visão, no séc. XIX, que permitiu demonstrar que o globo ocular direito está interligado ao hemisfério cerebral esquerdo; enquanto que o globo ocular esquerdo se relaciona com o hemisfério cerebral direito.

Atualmente tem-se verificado repetidamente que esta reversão não ocorre apenas com os nossos olhos, mas que, em verdade, o hemisfério cerebral esquerdo governa muitas das funções à direita do nosso organismo; enquanto que o hemisfério cerebral direito faz o mesmo em relação à nossa esquerda.

Esta reversão, envolvendo um órgão tão importante quanto o cérebro, traz conseqüências em muitas das manifestações dos seres humanos.

Uma criança, por exemplo, tem dificuldade em escrever as letras N ou E na sua forma correta, sendo muito freqüente, no seu início de aprendizado, desenhá-las na forma revertida ou . . . o que talvez seja uma conseqüência da forma como o seu inconsciente percebe o mundo.

No caso particular das letras, é curioso notar que o alfabeto cirílico — utilizado por cento e oitenta milhões de russos — possui os símbolos e , que são formas revertidas das letras R e N de nosso alfabeto.

Ainda no domínio da linguagem existem reversões, bastante curiosas, com certas sentenças, chamadas "palindrômicas", que tanto podem ser lidas no sentido normal de leitura como no sentido oposto, sem alterar o seu significado. Uma sentença deste tipo é latina e foi encontrada num dos muros da cidade de Pompéia, soterrada pela erupção do vulcão Vesúvio, no século I da nossa era (figura 2), cuja tradução tem significado (tanto se a leitura é feita no sentido tradicional, como no sentido oposto). Algo equivalente a: "o lavrador mantém cuidadosamente o arador no sulco".

sentido normal de leitura →

SATOR AREPO TENET OPERA ROTAS

← *sentido inverso de leitura*

Em português a sentença *palindrômica* mais conhecida é a que mostramos na figura 3. Em inglês, ocorrem frases como a da figura 4 que significa: "levante os lábios do aluno".

sentido normal de leitura →

SOCORRAM-ME SUBI NO ONIBUS EM MARROCOS

← *sentido inverso de leitura*

sentido normal de leitura →

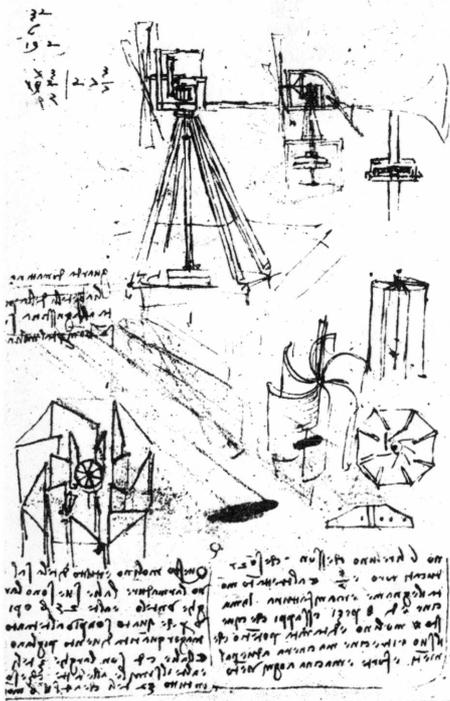
DRAW PUPIL'S LIP UPWARD

← *sentido inverso de leitura*

Contudo, a sentença mais curiosa continua sendo a latina pois, reescrita na forma "quadrada", como foi encontrada em Pompéia (figura 5) pode ser lida tanto nas linhas horizontais começando em cima, da esquerda para a direita, ou em baixo da direita para a esquerda, como nas linhas verticais, começando por cima, pela coluna da

SATOR
AREPO
TENET
OPERA
ROTAS

esquerda e lendo de cima para baixo, ou por baixo, pela coluna da direita e lendo de baixo para cima.

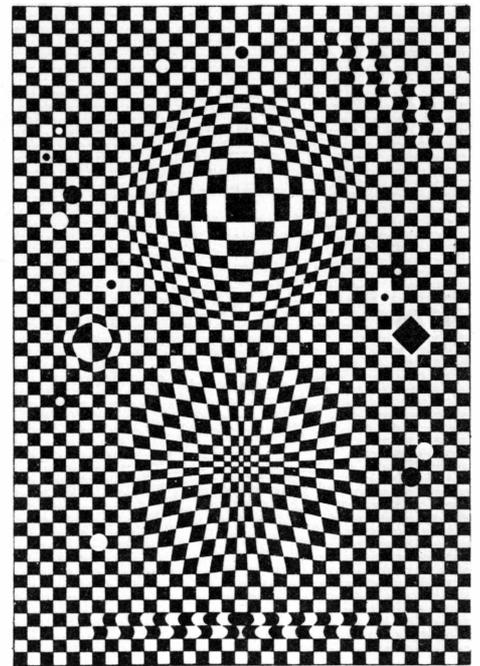
Finalmente, é curioso notar que Leonardo da Vinci (1452-1519), engenheiro, arquiteto, escultor e inventor italiano que, sob muitos aspectos se adiantou à sua época, descrevendo em pleno século XV, "máquinas voadoras", "paraquedas", "submarinos", "tanques", etc., redigiu as suas anotações em forma revertida, de tal maneira que o leitor só as entende ao lê-las refletidas num espelho plano (figura 6).

Mas a reversão não surge apenas no domínio da linguagem ou da escrita. Na música, por exemplo, também foi utilizada freqüentemente. Johann Sebastian Bach (1685-1750) em *Die Kunst Der Fuga (A Arte da Fuga)* compôs a Fuga-13 como imagem invertida da 12.

Joseph Haydn (1732-1809) na "Sonata para piano em *Lá*" compôs um minueto no qual a segunda parte é igual a primeira, mas tocada ao contrário.

Na música erudita moderna a reversão foi utilizada por Arnaldo Schonberg (1874-1951) que, em "Pierrot Lunaire", faz a música seguir até certo ponto e depois a faz voltar, repetindo as mesmas notas ao contrário.

Alunos de Schonberg, como Anton Webern (1883-1945) e Paul Hindemith (1895-1963) utilizam a reversão, na procura de novas harmonias na música. Em "Concerto para Nove Instrumentos — op. 24", Webern desenvolveu a partitura numa forma palindrômica equivalente ao quadrado achado em Pompéia, em *Ludus Tonalis* Hindermith escreveu em epílogo que é, com adição de um acorde final, igual ao prelúdio, só que tocado de cabeça para baixo e de trás para diante.

Além da escrita e da música, a reversão surgiu também na pintura, particularmente na obra de M.C. Escher (1898-1972), pintor holandês celebrizado em milhares de reproduções em cartazes, camisetas, etc. . . onde as suas estranhas simetrias e reversões são utilizadas sem a menor referência do autor (figura 7).

Num estilo diferente, o húngaro V. Vasalery (1908 —) também causou modismos marcantes nos padrões estéticos dos anos 70, utilizando constantes simetrias e reversões (figura 8).

Apesar de tudo isto, a reversão seria apenas uma curiosidade se de 1930 a 1960 não tivessem sido desenvolvidas uma série de observações sobre a estrutura da matéria, que sugeriam ser este fenômeno fundamental para o conhecimento do universo.

"Você gostaria de viver no País dos Espelhos, Kitty?
Será que, lá, lhe dariam leite?
Talvez o leite do País dos Espelhos
não seja bom para beber".
Alice, em Através do Espelho e o que Alice
encontrou lá, de Lewis Carroll.

```
L1
16514   2A 0C 40    LD HL, (D-FILE)  ; inicializa os registros. B é usado co-
16517   11 10 00    LD DE, $10         mo o contador de linhas; HL e DE in-
16520   19          ADD HL,DE          dicam o caractere da tela.
16521   54«LD D,H
16522   5D          LD E,L
16523   13          INC DE
16524   06 16       LD B, $16
L2
16526   7E          LD A, (HL)       ; armazena em A o caractere da tela e
16527   FE 76       CP $76             compara com o fim de linha. Se for
16528   28 09       JR Z, $09          fim de linha vai para L3.
16531   EB          EX DE,HL
16532   4E          LD C, (HL)       ; carrega (HL) em (DE) e (DE) com A.
16533   77          LD (HL), A
16534   EB          EX DE, HL
16535   71          LD (HL), C
L3
16536   2B          DEC HL           ; HL e DE são alterados e indicam os
16537   13          INC DE             PRÓXIMOS DOIS CARACTERES.
16538   18 F2       JR — 14
16540   11 31 00    LD DE, $31       ; quando chega ao final da linha, HL é
16543   19          ADD HL, DE         movimentado para a 16ª coluna da
16544   54          LD D, H            próxima linha e DE é carregado com e
16545   5D          LD E, L            valor de HL e incrementado.
16546   13          INC DE
16547   10 E9       DJNZ, — 23       ; repete o looping por 22 vezes.
16549   C9          RET                Ao terminar as 22 vezes, retorna ao
                                       BASIC.
```

Fig. 1

Você pode simular uma reversão na tela do seu computador com um curtíssimo programa em Linguagem de Máquina. Se você já tem algumas noções de Assembly, dê uma lida no programa com mnemônicos, nos códigos hexadecimais e nos comentários da figura 1.

Para entrar com este programa em seu computador, digite inicialmente em Basic o programa monitor de figura 2, lembrando que a linha 10 REM deve ter 36 pontos.

```
 10 REM ........................
............
 20 FOR X=16514 TO 16549
 30 SCROLL
 40 PRINT X,
 50 INPUT C
 60 PRINT C
 70 POKE X,C
 80 NEXT X
 90 SCROLL
100 PRINT "PRONTO"
```

Fig. 2

A função deste programa é introduzir os códigos de máquina (no caso já transformados de hexadecimal para decimal) na linha 10 REM através do POKE da linha 70. Ao digitar RUN, deverá aparecer o endereço 16514 na parte baixa da tela. A listagem da figura 3 mostra os códigos que você deverá digitar. Por exemplo, 42 e NEW LINE. Aparecerá então, o endereço 16515 e você deverá digitar o próximo código, no caso, 12 e assim sucessivamente.

```
LISTAGEM DECIMAL

16514..42        16515..12
16516..64        16517..17
16518..16        16519..0
16520..25        16521..84
16522..93        16523..19
16524..6         16525..22
16526..126       16527..254
16528..118       16529..40
16530..9         16531..235
16532..78        16533..119
16534..235       16535..113
16536..43        16537..19
16538..24        16539..242
16540..17        16541..49
16542..0         16543..25
16544..84        16545..93
16546..19        16547..16
16548..233       16549..201
```

Fig. 3

Ao terminar aparecerá um "pronto". Apague as linhas de 20 a 100 e digite as linhas de 20 a 60 como na figura 4.

```
20 INPUT A$
30 PRINT A$
40 PAUSE 300
50 RAND USR 16514
60 GOTO 20
```

Fig. 4

Ao rodar este novo programa (com a linha 10 do antigo), você deverá digitar uma palavra qualquer e pressionar NEW LINE. Cinco segundos depois (PAUSE 300), a sub-rotina em Linguagem de Máquina é chamada (RAND USR 16514) e a palavra digitada é escrita na tela em ordem revertida. Note que não se trata de uma verdadeira reversão, pois os símbolos continuam escritos da esquerda para a direita.

A frase em português citada neste artigo tem, por coincidência, uma linha de comprimento 32 (sem contar hífen de espaço). Experimente digitá-la.

```
 90 STOP
100 LIST
110 RAND USR 16514
120 FOR X=0 TO 20
130 NEXT X
140 GOTO 110
```

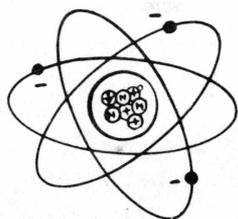
Fig.5

E veja o que é uma políndrome!

Você pode agora armazenar esta sub-rotina, muito útil principalmente na elaboração de jogos.

Experimente acrescentar ao seu programa as linhas da figura 5, digite RUN 100 e veja o efeito!

Pierlugi Piazzi

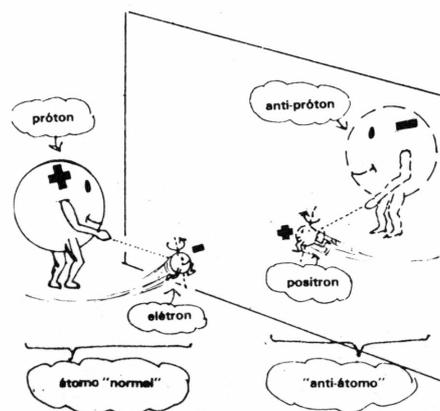
Por volta de 1930 aceitava-se que a matéria era constituída de átomos cuja estrutura lembrava, em muito, o nosso sistema solar (figura 9). No centro dessa estrutura se encontra o núcleo atômico, onde se concentra a maior parte da massa do átomo, formado por "prótons" de carga elétrica positiva e "neutrons", isentos de carga elétrica e, em torno do núcleo, os pequeníssimos elétrons, de carga elétrica negativa, orbitam como os planetas em torno do sol.

Em 1931, porém, esta curiosa estrutura mostrou-se insuficiente. Carl Anderson (1905 —), jovem físico norte-americano, estudando radiações de alta energia provenientes do espaço exterior, chamadas "radiações cósmicas", fotografou a trajetória de elétrons resultantes do choque destas radiações com partículas materiais, e observou que, ao submeter a região onde ocorriam os choques a um intenso campo eletromagnético, metade dos elétrons se desviavam num sentido, enquanto que a outra metade o fazia no sentido oposto.

A única maneira de explicar este comportamento levava a aceitar a existência de elétrons com carga elétrica positiva, que Anderson chamou de "pósitrons". A verificação da existência dos pósitrons deu a Anderson o prêmio Nobel de Física em 1936.

Em 1955, na Universidade da Califórnia, em Berkeley, o italiano Emílio Gino Segre (1905 —) e o norte-americano Owen Chamberlain (1920 —) determinaram a existência de prótons com carga elétrica negativa, motivo pelo qual compartilharam o prêmio Nobel de Física em 1959.

Como os pósitrons e os prótons negativos se apresentavam como imagens revertidas dos elétrons e prótons conhecidos até então, começou a se generalizar o costume de chamá-los de "anti-partículas".

No outono de 1956, foi determinada a existência de um reverso do neutron, o "anti-neutron" e, a partir daí, cada partícula descoberta no interior do átomo foi acompanhada de uma "anti-partícula" correspondente a sua imagem revertida num espelho plano.

As especulações em torno destes fatos levavam a acreditar na existência de "anti-átomos", e, por extensão, de "anti-moléculas" e mesmo de "anti-matéria" (figura 10).

Um átomo de matéria "normal" pode ser formado por um próton, de carga elétrica positiva e um elétron orbitando em torno dele, de carga elétrica negativa.

O "anti-átomo" correspondente, é a imagem revertida do átomo normal, com inversão do sinal das cargas elétricas.

De fato, anti-átomos foram detectados em laboratório, porém, desde o início se destacou a paridade destes elementos na natureza e seu pequeníssimo intervalo médio de vida, da ordem de 10^{-6}, uma vez que em média, neste exíguo intervalo de tempo, o anti-átomo colide com um átomo normal e, no resultado do choque, ambos se aniquilam, transformando-se em energia radiante.

Assim, uma pergunta foi surgindo e se impondo à medida que as anti-partículas eram detectadas:

— "Por que não ocorre uma distribuição simétrica de matéria e anti-matéria?"

Isto é, se as duas formas são possíveis no Universo, porque o nosso mundo é constituído por apenas uma delas?

Em 1956 os físicos Tsung Dao Lee (1926 —) e Chen Ning Yang (1922 —), no Instituto For Advanced Studies, em Princenton, mostraram, de forma imbatível, que na nossa região do Universo ocorre de fato uma orientação preferencial, com a conseqüente falha na simetria a nível de estrutura atômica. Esta demonstração valeu-lhes o prêmio Nobel de Física em 1957.

Contudo, é conveniente destacar que esta falha na simetria pode ser apenas local e que, a nível de Universo, poderão existir tanto uma orientação das partículas quanto a outra, distribuídas de tal forma que, enquanto alguns aglomerados de estrelas (galaxias) equivalentes a nossa região do Universo, são formados por um dos tipos de estrutura, outros, de mesmo número que os primeiros, são formados pelo segundo tipo. A este respeito tem-se apontado a galaxia conhecida como "Grande Nebulosa de Andrômeda", a aproximadamente $2x10^{20}$ km (isto é, em torno de duzentos quintilhões de quilômetros da Terra) que, ao ser observada através dos grandes telescópios, apresenta a forma simétrica e reversa da galaxia onde nos encontramos, como sendo uma candidata razoável para formar conosco um par "matéria" (nós) e "anti-matéria" (andrômeda). Assim, não deixa de ser curioso pensar que talvez neste momento, numa "anti-Terra" de Andrômeda, um "anti-você" esteja lendo um "anti-texto" equivalente ao que você está lendo agora, com tanta "anti-surpresa", "anti-admiração", e "anti-espanto", quanto a sua própria surpresa, admiração e espanto.

Contudo, se algum dia você vier a encontrar teu "anti-você", evite qualquer contato direto, pois a experiência mostra que as respectivas massa e "anti-massa" se transformariam imediatamente numa quantidade imensa de energia radiante, que provocaria nas vizinhanças, uma catástrofe equivalente ao impacto direto de um moderno míssil nuclear!

Deixando as catástrofes atômicas de lado, observamos ainda que o fato de existirem átomos e moléculas revertidas deveria gerar seres vivos, constituídos por moléculas complexas (proteínas) revertidas, mas que, mais uma vez, como aconteceu com a matéria e antimatéria, na região do Universo que habitamos ocorre preferência por uma determinada orientação e sentido de

rotação das cadeias moleculares, em relação a orientação e rotação opostas.

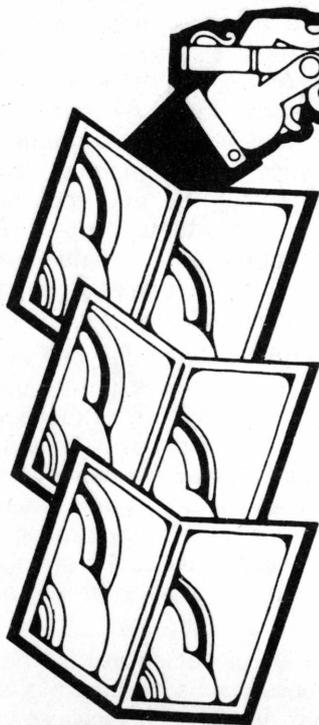
Desta forma, na nossa região do Universo (na Terra, pelo menos), as proteínas formadoras de todos os seres vivos, das amebas aos homens, são exclusivamente "levógiras", isto é, estão orientadas como a mão esquerda, ocorrendo proteínas "dextrógiras", isto é, orientadas como a mão direita somente em compostos artificiais, obtidos a nível de laboratório.

A este respeito vale destacar a curiosa observação efetuada por Charles L. Dodgson (1832-1898), que sob o pseudônimo de Lewis Carroll, escreveu em 1871, um clássico dos contos de fadas: "Através dos espelhos, e o que Alice encontrou lá". Matemático da melhor qualidade, Dodgson fez, no meio do relato das aventuras de Alice, uma indagação que muito tem intrigado, inclusive os físicos, até os dias de hoje.

A certa altura dos acontecimentos Alice olha na direção de um espelho plano, e, vendo a sua imagem refletida segundo o copo de leite que no momento tomava, se pergunta (em 1871!) se o leite do espelho seria bom de beber.

Pensou-se durante algum tempo que tal leite não seria digerido porque as enzimas do corpo humano, destinadas a agir sobre as moléculas "levógiras", não seriam capazes de assimilar as moléculas revertidas do espelho. E dessa forma, Alice teria um tremendo desarranjo intestinal!

Mas, na verdade, o caso poderia ser bem pior, pois o leite do espelho seria de fato "anti-leite", e assim que os delicados lábios de Alice encostassem na superfície do "leite do espelho", uma tremenda explosão, resultando do aniquilamento matéria-antimatéria, varreria toda a cidade de Londres, transformando-a num deserto calcinado, muito antes de nossa inocente heroina pode sofrer qualquer dor de barriga! (Figura 11).

O prof. ***Álvaro Csapo Talavera*** é engenheiro eletricista formado pela Escola de Engenharia Mauá fez curso no Instituto de Física da USP atualmente é professor de Física do Curso Anglo-Latino e co-autor do "PROJETO 2º grau" editado pela MARCO EDITORIAL, do qual foi extraído o texto deste artigo.

COMO FAZER SUA ASSINATURA

Para obter seu exemplar mensal (12 números) da revista **Microhobby** contendo muitos programas para o TK como também para o Apple e o TRS-80, inúmeras dicas e as últimas novidades na área de informática, você deve fazer uma assinatura.

É importante ressaltar que o recebimento da revista é considerado a partir da data de recebimento do pedido de assinatura, porém, há um período de 30 dias de *carência* até a revista chegar às suas mãos. Os números anteriores da revista podem ser adquiridos se as tivermos no momento do pedido em nossos estoques, através do telefone **255-0722**, diretamente com o Departamento Comercial.

Para ter acesso a todas estas vantagens basta preencher, corretamente, o cupom anexo, colocá-lo no correio junto a um cheque nominal ou vale postal em nome de Micromega Publicações e Material Didático, no valor de **Cr$ 19.800,00**.

O envelope deverá ser selado e endereçado com os seguinte dizeres:

MICROMEGA P.M.D. LTDA.
Departamento de Assinaturas
Caixa Postal 54096
CEP 01296 – São Paulo, SP.

e dentro dele não deverá ter nada além do cheque e o cupom.

No verso do cheque, escreva:

"Destina-se ao pagamento de uma assinatura (12 números) da revista Microhobby".

Quando este cheque for devolvido ao seu Banco com nosso endôsso, servirá (para você), de comprovante provisório até que nosso recibo seja confeccionado e enviado pelo correio.

ANUNCIE

NA EDIÇÃO EXTRA DA REVISTA

MICROHOBBY

A edição extra da MICROHOBBY trará uma retrospectiva da informática no Brasil, entrevistas com quem entende da matéria, grande número de programas para a linha TK (TK83/85 e TK2000-color), TRS-80, APPLE e compatíveis, lançamentos de software, periféricos e equipamentos e também uma completa lista de software e equipamentos existentes no mercado

Cobertura de todos expositores da Feira de Informática, 84 — RJ.

Anuncie já os seus produtos ou serviços nessa edição extra da revista MICROHOBBY, que terão um retorno garantido através de

120.000 *leitores em todo País.*

Para anunciar, basta telefonar.

(011) 255-0722

FECHAMENTO PARA PUBLICIDADE: 30/09/84

Leoni

MICROHOBBY, A REVISTA QUE PÕE VOCÊ EM DIA COM A INFORMÁTICA.

Uma publicação da
MICROMEGA Publ. e Mat. Did. Ltda.
Av. Angélica, 2.318 — 13º andar
Cxa. Postal 54096 — CEP 01296 — S. Paulo — SP

DINHEIRO NO TEMPO

2ª parte

Luiz Tarcísio de Carvalho Jr

A) *PROGRAMA 1*

1º OPÇÃO — Calcula o VALOR PRESENTE, conhecidos: o número de períodos, taxa de juros por período e o VALOR FUTURO.

Exemplo: Qual o valor que deve ser investido hoje, a 12 por cento ao mês, para que, daqui a doze meses, possam ser resgatados Cr$ 100.000,00 num papel de renda fixa?

Entrando com a taxa de juros (12%), o número de períodos (12) e o valor futuro (Cr$ 100.000,00) o programa deve apresentar a resposta conforme a figura 2.

2ª OPÇÃO — Calcula o VALOR PRESENTE, conhecidos: o número de períodos, taxa de juros por períodos e o VALOR DA PRESTAÇÃO.

Exemplo: Um anúncio diz que um aparelho pode ser comprado em 24 meses com uma prestação de Cr$ 18.000,00, sem entrada. Sabendo que a loja cobra 10 por cento de juros ao mês, qual o preço do aparelho à vista?

O programa deve responder conforme a figura 3.

3ª OPÇÃO — Calcula o VALOR FUTURO, conhecidos: o número de períodos, a taxa de juros por períodos e o VALOR PRESENTE.

Exemplo: Se forem investidos Cr$ 800.000,00, hoje, a uma taxa de juros mensal de 12 por cento, qual deverá ser o montante daqui a um ano?

O programa deve responder conforme a figura 4.

4ª OPÇÃO — Calcula o VALOR FUTURO, conhecidos: o número de períodos, a taxa de juros por períodos e o VALOR DA PRESTAÇÃO.

Exemplo: Se depositarmos Cr$ 15.000,00 todo o mês numa instituição financeira que paga 6 por cento de juros ao mês, qual o montante após dois anos de depósitos?

O programa deve responder conforme a figura 5.

5ª OPÇÃO — Calcula o VALOR DA PRESTAÇÃO, conhecidos: o número de períodos, a taxa de juros por períodos e o VALOR PRESENTE.

Exemplo: Qual o valor da prestação de um automóvel se ele custa Cr$ 4.000.000,00, foi financiada em 36 meses a 10 por cento ao mês de juros?

O programa deve responder conforme a figura 6.

6ª OPÇÃO — Calcula o VALOR DA PRESTAÇÃO conhecidos: o número de período, a taxa de juros por períodos e o VALOR FUTURO.

Exemplo: Uma reportagem diz que um novo microcomputador será lançado daqui a dez meses ao preço de Cr$ 600.000,00. Quanto deve ser depositado mensalmente numa caderneta de poupança que paga 8 por cento ao mês, para que se consiga a quantia para comprar o aparelho quando for lançado?

O programa deve responder conforme a figura 7.

OBSERVAÇÃO: Após cada cálculo o ***Programa 1*** oferece invariavelmente as seguintes opções:

1) Mudar de opção no programa 1, teclando A.

2) Manter a mesma opção com outros valores (tecla B), com mesma taxa de juros (tecla C) ou com a mesma taxa de juros e mesmo número de períodos (tecla D).

3) Mudar para o ***programa 2,*** teclando I (neste caso, você voltará ao quadro inicial de escolhas).»

4) Parar a execução do programa. Veja a figura 8.

```
QUAL A TAXA DE JUROS EXPRESSA EM
PORCENTAGEM?
J=12 POR CENTO

QUAL O NUMERO DE PERIODOS?
N=12

QUAL O VALOR FUTURO?
F=CR$ 100.000,00

VALOR PRESENTE=CR$ 25.667,51
```

Figura 2

```
QUAL A TAXA DE JUROS EXPRESSA EM
PORCENTAGEM?
J=10 POR CENTO

QUAL O NUMERO DE PERIODOS?
N=24

QUAL O VALOR DA PRESTACAO?
A=CR$ 18.000,00

VALOR PRESENTE=CR$ 161.725,39
```

Figura 3

```
QUAL A TAXA DE JUROS EXPRESSA EM
PORCENTAGEM?
J=12 POR CENTO

QUAL O NUMERO DE PERIODOS?
N=12

QUAL O VALOR PRESENTE?
P=CR$ 800.000,00

VALOR FUTURO=CR$ 3.116.780,80
```

Figura 4

```
QUAL A TAXA DE JUROS EXPRESSA EM
PORCENTAGEM?
J=6 POR CENTO

QUAL O NUMERO DE PERIODOS?
N=24

QUAL O VALOR DA PRESTACAO?
A=CR$ 15.000,00

VALOR FUTURO=CR$ 762.233,66
```

Figura 5

```
QUAL A TAXA DE JUROS EXPRESSA EM
PORCENTAGEM?
J=10 POR CENTO

QUAL O NUMERO DE PERIODOS?
N=36

QUAL O VALOR PRESENTE?
P=CR$ 4.000.000,00

PRESTACAO=CR$ 413.372,26
```

Figura 6

```
QUAL A TAXA DE JUROS EXPRESSA EM
PORCENTAGEM?
J=8 POR CENTO

QUAL O NUMERO DE PERIODOS?
N=10

QUAL O VALOR FUTURO?
F=CR$ 600.000,00

PRESTACAO=CR$ 41.417,69
```

Figura 7

Figura 8

B) *PROGRAMA 2*

OPÇÃO 1 — Calcula a TAXA DE JUROS, conhecidos o VALOR PRESENTE, o VALOR DA PRESTAÇÃO e o número de períodos.

Exemplo: Um estabelecimento comercial financiou a quantia de Cr$ 180.000,00 em 6 meses. A prestação pedida foi de Cr$ 45.000,00. Qual a taxa mensal de juros cobrada pelo estabelecimento?

O programa responderá conforme a figura 9.

OPÇÃO 2 — Calcula a TAXA DE JUROS, conhecidos: o VALOR PRESENTE, o VALOR FUTURO e o número de períodos.

Exemplo: Um terreno foi comprado por Cr$ 1.000.000,00 numa certa data e passou a valer Cr$ 2.100.000,00 um ano depois. Qual foi a taxa mensal de valorização do terreno?

Fazendo o preço inicial do terreno ser o valor presente e o preço final ser o valor futuro, 12 meses depois, o programa responderá conforme a figura 10.

OPÇÃO 3 — Calcula o NÚMERO DE PERÍODOS, conhecidos: o VALOR PRESENTE, o VALOR DA PRESTAÇÃO e a taxa de juros por período.

Exemplo: Qual o número de prestações que uma pessoa deverá pagar para saldar um empréstimo de Cr$ 200.000,00 se ela só pode pagar no máximo a prestação de Cr$ 32.000,00, sabendo-se que a taxa de juros ao mês é de 8 por cento?

O programa responderá conforme a figura 11.

OPÇÃO 4 — Calcula o NÚMERO DE PERÍODOS, conhecidos: o VALOR FUTURO, o VALOR DA PRESTAÇÃO e a taxa de juros por período.

Exemplo: Durante quantos meses deve-se depositar a quantia de Cr$ 8.000,00 para poder-se resgatar a quantia de Cr$ 380.000,00, com uma taxa de juros mensal de 10 por cento?

O programa responderá conforme a figura 12.

OPÇÃO 5 — Calcula o NÚMERO DE PERÍODOS, conhecidos: o VALOR PRESENTE, o VALOR FUTURO e a taxa de juros por período.

Exemplo: Quantos meses deve-se esperar para que o capital de Cr$ 5.000.000,00 triplique a uma taxa de juros mensal de 7,5 por cento?

O programa fornecerá a resposta de acordo com a figura 13.

OPÇÃO 6 — Esta opção fornece o SALDO DEVEDOR após o pagamento de uma dada prestação (L). Deve-se conhecer o VALOR PRESENTE, o VALOR DA PRESTAÇÃO, a TAXA DE JUROS por período e o número da prestação (L) após à qual quer-se saber o saldo devedor.

Exemplo: João financiou a quantia de Cr$ 217.800,00 e paga uma prestação mensal de Cr$ 23.000,00 com uma taxa de juros de 8,5 por cento ao mês. Após um ano de pagamento (12 prestações pagas) de quanto dinheiro João precisa para quitar a dívida?

O programa deve responder de acordo com a figura 14.

OBSERVAÇÃO: Após cada cálculo, ***o programa 2*** oferece invariavelmente as seguintes opções:

1) Mudar de opção no programa 2, teclando ***A;***

2) Manter a mesma opção com outros valores, teclando ***B;***

3) Mudar para o programa 1, teclando ***I*** (neste caso, você volta ao quadro inicial);

4) Parar a execução do programa. Veja a figura 15.

III — *Entrada e Saída de Dados*

A entrada de dados no programa é feita segundo os padrões de Aritmética

Financeira. Para tal, são utilizadas duas sub-rotinas (5120 e 5220) que foram adaptadas de uma sub-rotina de Flavio Rossini (ver número 3 de Microhobby). Assim sendo, as grandezas TAXA DE JUROS, VALOR PRESENTE, VALOR FUTURO e VALOR DE PRESTAÇÃO, devem ser colocados no programa da maneira como costumamos escrevê-las, ou seja, em sua notação decimal, devem ser escritas com ***vírgula e não com ponto.***

Exemplo: Se uma taxa de juros é de sete e meio por cento, ela deve ser inserida como 7,5 e não como 7.5. Se você inserir 7.5, o computador "entenderá" esse valor como 75, pois as sub-rotinas mencionadas fazem o computador ignorar os pontos.

A saída dos dados é "filtrada" por três sub-rotinas: as de número 4670 e 4720, que arredondam os resultados, e a de número 4870, que gera saída em formato financeiro. Esta última também foi adaptada de uma sub-rotina de Flavio Rossini (edição número 3 de Microhobby).

```
QUAL O VALOR PRESENTE?
P=CR$ 180.000,00

QUAL A PRESTACAO?
A=CR$ 45.000,00

QUAL O NUMERO DE PERIODOS?
N=6

TAXA DE JUROS=12,96 POR CENTO
```

Figura 9

```
QUAL O VALOR PRESENTE?
P=CR$ 200.000,00

QUAL A PRESTACAO?
A=CR$ 32.000,00

QUAL A TAXA DE JUROS EXPRESSA EM
PORCENTAGEM?
J=8 POR CENTO

NUMERO DE PERIODOS=9,01
```

Figura 11

```
QUAL O VALOR PRESENTE?
P=CR$ 1.000.000,00

QUAL O VALOR FUTURO?
F=CR$ 2.100.000,00

QUAL O NUMERO DE PERIODOS?
N=12

TAXA DE JUROS=6,38 POR CENTO
```

Figura 10

```
QUAL O VALOR FUTURO?
F=CR$ 380.000,00

QUAL A PRESTACAO?
A=CR$ 8.000,00

QUAL A TAXA DE JUROS EXPRESSA EM
PORCENTAGEM?
J=10 POR CENTO

NUMERO DE PERIODOS=18,35
```

Figura 12

```
QUAL O VALOR PRESENTE?
P=CR$ 500.000,00

QUAL O VALOR FUTURO?
F=CR$ 1.500.000,00

QUAL A TAXA DE JUROS EXPRESSA EM
PORCENTAGEM?
J=7,5 POR CENTO

NUMERO DE PERIODOS=15,19
```

Figura 13

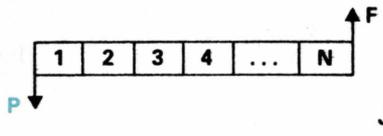
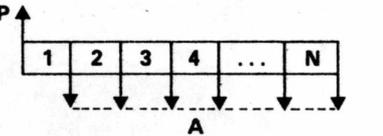
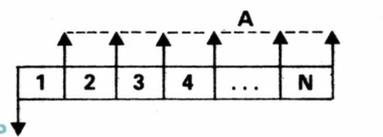
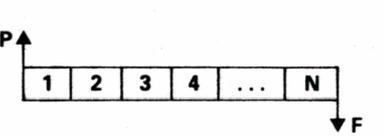
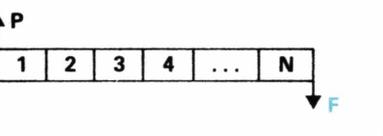
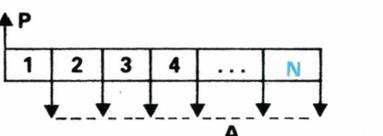
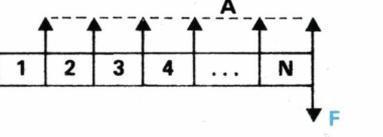
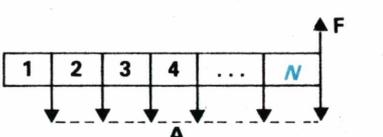
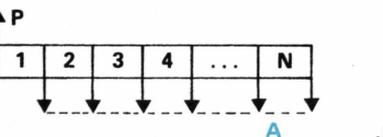
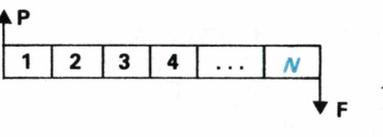
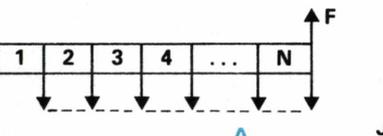
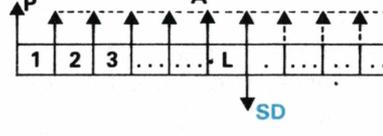
```
QUAL A TAXA DE JUROS EXPRESSA EM
PORCENTAGEM?
J=8,5 POR CENTO
QUAL O VALOR PRESENTE?
P=CR$ 217.800,00
QUAL O VALOR DA PRESTACAO?
A=CR$ 23.000,00
APOS QUAL PRESTACAO VOCE QUER
SABER O SALDO DEVEDOR?
L=12

SALDO DEVEDOR=CR$ 130.082,52
```

Figura 14

Figura 15

QUADRO RESUMO DAS FUNÇÕES DO PROGRAMA

	PROGRAMA 1	PROGRAMA 2
OPÇÃO 1	DADOS: J, N, F → calcula *P*	DADOS: P, A, N → calcula *J*
OPÇÃO 2	DADOS: J, N, A → calcula *P*	DADOS: P, F, N → calcula *J*
OPÇÃO 3	DADOS: J, N, P → calcula *F*	DADOS: P, A, J → calcula *N*
OPÇÃO 4	DADOS: J, N, A → calcula *F*	DADOS: F, A, J → calcula *N*
OPÇÃO 5	DADOS: J, N, P → calcula *A*	DADOS: P, F, J → calcula *N*
OPÇÃO 6	DADOS: J, N, F → calcula *A*	DADOS: P, A, J, L → calcula *SD*

IV — *Digitação do Programa*

A listagem do programa, visto a diversidade de opções que ele oferece, é muito extensa. Recomendamos que você a faça em modo FAST e que a realize por partes, sempre gravando em fita a parte já digitada.

Assim, os acidentes, como a falta de energia elétrica, não serão tão catastróficos.

Boa sorte! ○

O AMOR DO Sr. NABOR

Renato da Silva Oliveira

"Eu, Tolen e Ramarujan saimos da India há quatro dias. Deixamos o Dalai Lama em Dehra Dun e lá, decidimos visitar um antigo amigo nosso na Inglaterra.

Estamos indo por mar, num navio de nome GANDHI que transporta quase dois mil passageiros a bordo.

Nossa estadia no GANDHI está sendo mais agradável do que esperávamos, pois nos três camarotes em frente aos nossos, estão alojados três belas e jovens passageiras.

Nós as conhecemos na noite passada, pouco antes do jantar. Para nossa sorte, uma delas tropeçou numa emenda do carpete do salão de jantar, derrubou o garçom que nos servia e acabou caindo nos braços de Tolen.

Após nos apresentarmos, insistimos para que elas se sentassem à mesa conosco e, diante de nossa persistência, o convite acabou sendo aceito.

Não foi preciso conversarmos muito para percebermos a extraordinária coincidência que ocorrera: a moça que tropeçou no carpete era brasileira e as outras duas eram de nacionalidades hindú e chinesa. Aliás, a chinesa nascera em Lhasa, no Tibet.

Com poucos argumentos, conseguimos convencê-las a nos acompanhar no jantar preparado especialmente para nós com iguarias que Ramarujan trouxera de Ajshrangivad, onde ele mesmo as cultivara.

Logo ficamos sabendo que as moças haviam se conhecido em Nova Delhi e que, por outra coincidência não menos extraordinária do que a que estava ocorrendo, iam todas à Inglaterra. Por isso, tinham resolvido viajar juntas.

Enquanto ***Ramarujan*** *bebia calado seu suco de morangos silvestres puro e concentrado,* Tolen entretia duas das moças e eu conversava com a outra. Ela me dizia que uma delas era *programadora, outra era matemática e outra secretária executiva.*

Eu estava com pouca fome e em meu prato haviam apenas quatro morangos silvestres. Notei que uma das moças também não estava comendo quase nada, mas em compensação, bebia suco de morangos quase sem parar.

Não sei bem porque, mas senti-me atraído por essa moça. Achei que ela também estava demonstrando interesse por mim e resolvi começar uma conversa qualquer.

Falamos de muitas coisas e eu surpreendi-me com seus conhecimentos.

Até o final do jantar todos falamos sempre em inglês e, talvez por isso, pouco antes de deixarmos a mesa, *a chinesa confidenciava-se despreocupadamente em Latin com a matemática. Possivelmente,* elas imaginavam que nós não estávamos entendendo nada.

Elas observavam que *"o cavalheiro" que bebia suco de morangos com leite tinha em seu prato exatamente três vezes mais morangos que a secretária, e que o "cavalheiro" de mesma nacionalidade que ela não estava bebendo nada.*

Nesse instante, só para deixá-las ruborizadas, Tolen perguntou a elas em *Latin* quais outras línguas elas falavam. As duas moças coraram enquanto nos levantamos. Desculparam-se e caminhamos todos para nossos aposentos. Antes de lá chegarmos, a moça hindú e a programadora foram à "toilette" e, enquanto às aguardávamos, convidei a moça pela qual me interessei para irmos ao convés observar a noite. O que aconteceu a partir daí foi uma das coisas mais inesperadas para mim. Creio que me apaixonei."

Obviamente, você já sabe a quem pertence o texto acima. Pois bem, o quebra-cabeça desta edição consiste em fazer um programa para que o TK descubra *qual o amor do Sr. Nabor Rosenthal* (a hindú, a brasileira ou a chinesa e ainda, qual a profissão e respectiva nacionalidade de cada uma das moças!)

TEMPESTADE

Memória ocupada — 13 29 bytes
Soma Sintática — 25991

Gustavo Egídio de Almeida

Muitos leitores que adquirem um micro em tempo recente, e que estão apenas engatinhando na linguagem Basic, dão preferência ao digitar programas simples e curtos.

Apresento-lhes agora um programa que ocupa apenas 2k de memória: **Tempestade.** (Figura 1)

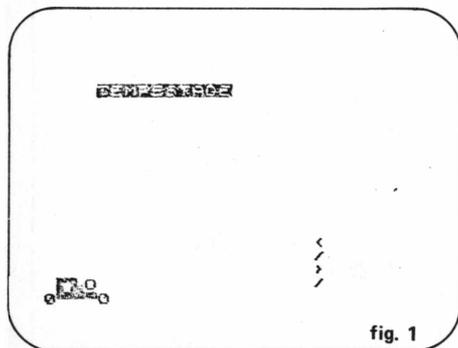



fig. 1

Neste jogo, em meio a uma tenebrosa tempestade, você está fugindo de um castelo mal assombrado, na qual seres que habitam aquele local, como morcegos e outros terríveis monstros terrestres estão em sua perseguição.

No decorrer da fuga (por sorte ou não), você encontra um veículo conversível estacionado.

Por medida de segurança, você tentará fechar o capô do carro, mas ao tentar este feito, as ferragens deste enguiça e, como você não dispõe de muito tempo para consertá-lo, acaba por dirigir o veículo do jeito que está. Porém, a tempestade é terrível e você precisará de muita sorte para desviar-se dos raios. (fig. 2)

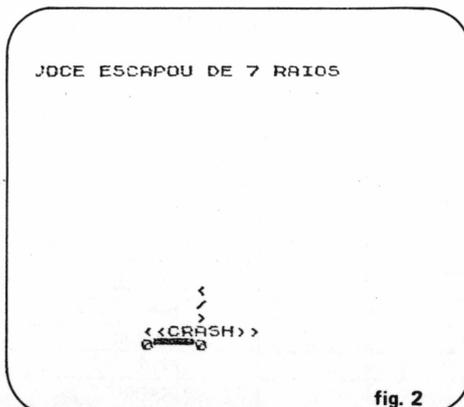



fig. 2

Para dirigir seu carro use as teclas:

Ø — Para esquerda

1 — Para direita

Como o programa é curto, algumas pequenas observações devem ser feitas, como:

Gráficos usados nas linhas 4Ø e 5Ø:

4Ø — 2 SPACES, SHIFT GRAPHIC E, SHIFT GRAPHIC W, Ø, 1 SPACE

5Ø — 1 SPACE, Ø, 3 GRAPHIC 7, Ø, 1 SPACE

```
  10 PRINT AT VAL "5",VAL "5";"TEMPESTADE"
  16 LET L=PI-PI
  20 LET A=VAL "19"
  30 LET B=PI-PI
  40 PRINT AT A,B;"  ▛▖O "
  50 PRINT AT A+VAL "1",B;" O▀▀▀O "
  60 LET C=INT (RND*VAL "31")+VAL "1"
  70 LET D=VAL "16"
  80 PRINT AT D,C;"<"
  90 PRINT AT D+VAL "1",C;"/"
 100 PRINT AT D+VAL "2",C;">"
 105 PRINT AT D+VAL "3",C;"/"
 106 LET L=L+VAL "1"
 110 IF INKEY$="0" THEN LET B=B+VAL "1"
 120 IF INKEY$="1" THEN LET B=B-VAL "1"
 130 IF C=B+VAL "4" OR C=B+VAL "2" OR C=B+VAL "3" OR C=B+VAL "1" THEN GOTO VAL "200"
 140 CLS
 150 GOTO VAL "40"
 200 PRINT AT A,B;"<<CRASH>>"
 210 PRINT AT VAL "0",VAL "0";"VOCE ESCAPOU DE ";L-VAL "1";" RAIOS"
```

TK 2000

texto: Nancy Mitie Ariga
programa: Sandra C. Van Blarcum de Graaff

Traçados gráficos em alta resolução

Traçados gráficos em alta resolução

Desenhar polígonos em perspectiva com uma boa resolução gráfica é um dos grandes sonhos de todo o programador de computador. Isto porque a maioria dos microcomputadores não possui uma boa definição gráfica.

Entretanto, o TK-2000 e o Apple possuem uma página em alta resolução que possibilita uma definição de um número maior de pontos na tela, se compararmos com a definição dos pontos no modo normal. Esta alta definição, permite um traçado com mais detalhes.

Comando HGR e TEXT

O comando HGR permite que o TK-2000 e o APPLE entrem no modo gráfico em alta resolução. Esta alta resolução permite a definição de 280 pontos horizontais e 160 verticais, num total de 44800 pontos e as últimas duas linhas finais da tela permanecem no modo texto.

Para retornar ao modo texto utiliza-se o comando TEXT.

Comando HGR2

Comando HGR2 permite uma alta resolução gráfica em toda a tela de 280 pontos horizontais e 192 pontos verticais num total de 53760 pontos.

Comando HPLOT

HPLOT X, Y é a forma básica desta instrução que permite que um ponto na coordenada horizontal X e coordenada vertical Y seja colocado na tela.

Porém, se utilizarmos esta instrução na forma HPLOT X1, Y1 TO X2, Y2 será traçado uma reta dos pontos X1, Y1 ao X2, Y2. Esta forma de instrução facilita o traçado de polígonos fechados. Por exemplo, se digitarmos uma instrução na forma HPLOT X1, Y1 TO X2, Y1 TO X2, Y2 TO X1, Y2 TO X1,Y1 teremos um polígono fechado na forma retangular.

Comando HCOLOR

A forma deste comando deve ser:
HCOLOR = x

onde X corresponde a um número de 1 a 7 conforme a tabela na figura 1.

Esta instrução só pode ser utilizada no modo em alta resolução.

Poligonos em alta resolução

O programa da figura 2 é um programa que permite o traçado de polígonos que formarão uma figura em perspectiva como a figura 3.

Digite o programa da figura 2 e depois digite RUN para começar a rodar o programa.

A linha 2Ø espera que você introduza um número N que será o número de polígonos que serão desenhados na tela de cada lado (N polígonos à direita e N polígonos à esquerda). Portanto quanto maior o número N, maior será o número de polígonos a serem traçados e menor será o espaço entre cada polígono (veja a diferença comparando as figuras 4 e 5).

A linha 3Ø permite a entrada no modo gráfico em alta resolução de toda a tela.

A linha 4Ø indica a cor que será o polígono. Caso você prefira modificar a cor, veja a tabela de cores em alta resolução para o TK-2000 e para o Apple na figura 1.

As linhas 5Ø e 12Ø contem as coordenadas dos dois primeiros polígonos a serem formados na parte inferior da tela:

T1 e T2 representam as abscissas do primeiro polígono da direita
T3 e T4 representam as ordenadas do primeiro polígono da direita
T5 e T6 representam as abscissas do primeiro polígono da esquerda
T7 e T8 representam as ordenadas do primeiro polígono da esquerda

TK 2000		APPLE	
Número	Cor	Número	Cor
0	preto	0	preto
1	azul	1	verde
2	verde	2	violeta
3	branco	3	branco
4	azul	4	preto
5	vermelho	5	laranja
6	cyan	6	azul
7	branco	7	branco

Fig. 1

```
10  REM  ***** POLIG
ONOS TRIDIMENSIONA
 *****
20  HOME : INPUT "NU
MERO DE PASSOS ?";
N
30  HGR2
40  HCOLOR  = 7
50 T1 = 150
60 T2 = 270
70 T3 = 95
80 T4 = 190
90 T5 = 10
100 T6 = 130
110 T7 = 95
120 T8 = 190
130  HPLOT T1,T3 TO
T2,T3 TO T2,T4 TO
T1,T4 TO T1,T3
140  HPLOT T5,T7 TO
T6,T7 TO T6,T8 TO
T5,T8 TO T5,T7
150 X = 279:Y = 0
160 X1 = 0
170  FOR I = 1 TO N
180  IF I < 25 THEN
X = X - (278 / 25)
:X1 = X1 + (278 / 25
): GOTO 220
190  IF I = 25 THEN
X = 0:X1 = 279: GOTO
220
200  IF I < 50 THEN
X = X + (278 / 25)
:X1 = X1 - (278 / 25
): GOTO 220
210  IF I = 50 THEN
X = 279:X1 = 0: GOTO
220
220 Q1 = ((N - I) *
T1 + I * X) / N
230 R1 = ((N - I) *
T5 + I * X1) / N
240 Q2 = ((N - I) *
T2 + I * X) / N
250 R2 = ((N - I) *
T6 + I * X1) / N
260 Q3 = ((N - I) *
T3 + I * Y) / N
270 R3 = ((N - I) *
T7 + I * Y) / N
280 Q4 = ((N - I) *
T4 + I * Y) / N
290 R4 = ((N - I) *
T8 + I * Y) / N
300 Q5 = ((N - I) *
X + I * T1) / N
310 R5 = ((N - I) *
X1 + I * T5) / N
320 Q6 = ((N - I) *
X + I * T2) / N
330 R6 = ((N - I) *
X1 + I * T6) / N
340 Q7 = ((N - I) *
Y + I * T3) / N
350 R7 = ((N - I) *
Y + I * T7) / N
360 Q8 = ((N - I) *
Y + I * T4) / N
370 R8 = ((N - I) *
Y + I * T8) / N
380  HPLOT Q1,Q3 TO
Q2,Q3 TO Q2,Q4 TO
Q1,Q4 TO Q1,Q3
390  HPLOT Q5,Q7 TO
Q6,Q7 TO Q6,Q8 TO
Q5,Q8 TO Q5,Q7
400  HPLOT R1,R3 TO
R2,R3 TO R2,R4 TO
R1,R4 TO R1,R3
410  HPLOT R5,R7 TO
R6,R7 TO R6,R8 TO
R5,R8 TO R5,R7
420  IF I >  = N / 2
 THEN 440
430  NEXT I
440  GET S$
450  TEXT
470  GOTO 20
```

Fig.2

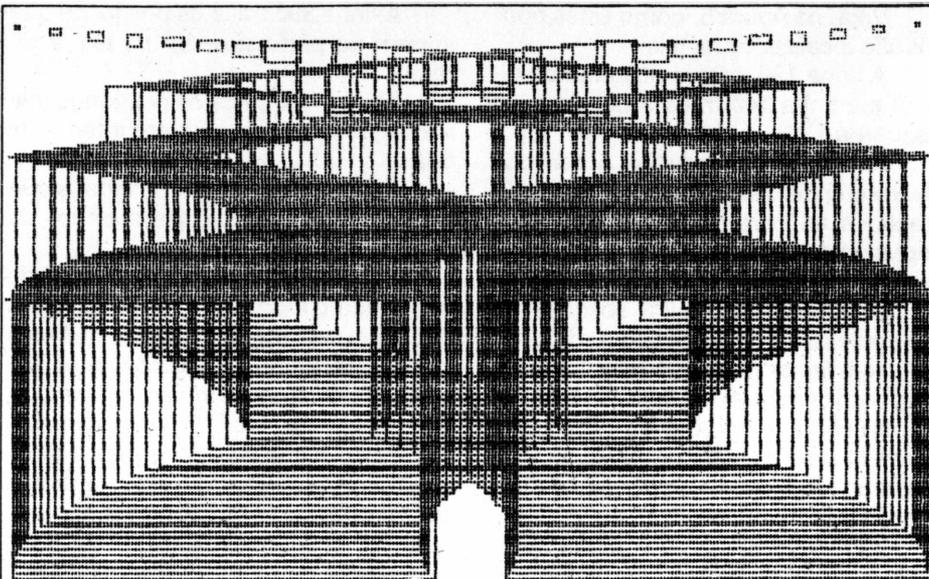



Fig. 3

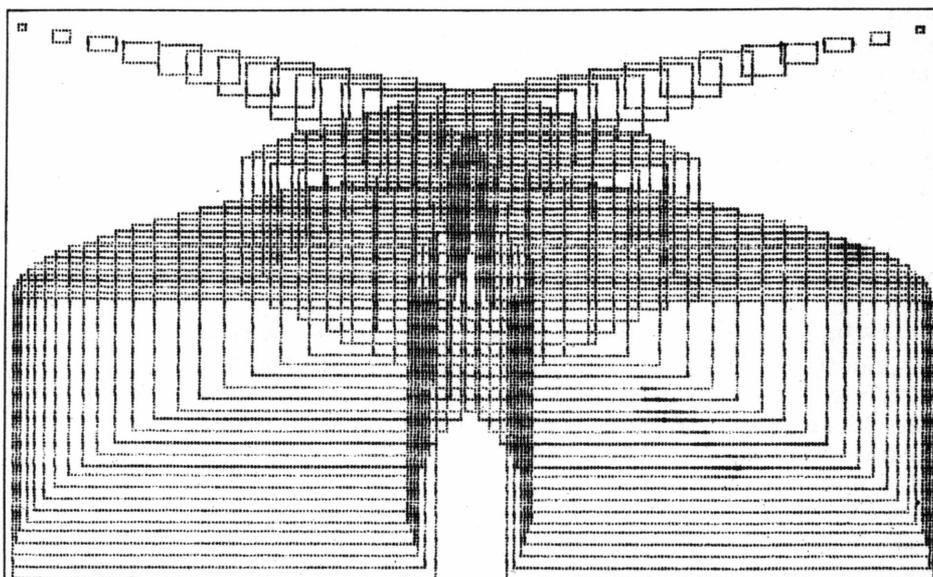



Fig. 4

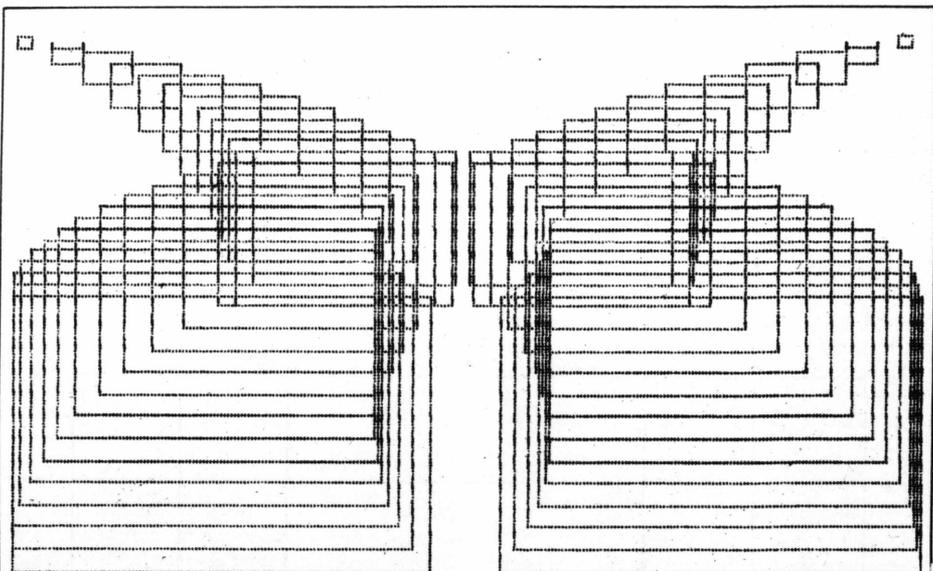



Fig. 5

Veja, na figura 6, como estes pontos são localizados na tela.

A linha 130 traça o polígono da direita e a linha 140 traça o polígono da esquerda.

A linha 150 possui as ordenadas de referência dos polígonos que são traçados na parte superior da tela. X representa a abscissa do polígono à direita e Y representa a ordenada.

A linha 160 possui a abscissa X1 do primeiro polígono da esquerda.

Nas linhas 180 a 210 são definidos os pontos em que o polígono deve mudar o sentido de formação do prisma, fazendo uma curva. A cada 25 polígonos traçados muda-se o sentido de formação.

Nas linhas 220 e 370 estão as coordenadas dos polígonos que serão formados:
Q1 e Q2 representam as abscissas dos polígonos que iniciam sua formação na parte inferior da tela à direita;
Q3 e Q4 representam as ordenadas dos polígonos que iniciam sua formação na parte inferior da tela à direita;
Q5 e Q6 representam as abscissas dos polígonos que iniciam sua formação na parte superior da tela à direita;
Q7 e Q8 representam as ordenadas dos polígonos que iniciam sua formação na parte superior da tela à direita;
R1 e R2 representam as abscissas dos polígonos que iniciam sua formação na parte inferior da tela à esquerda;
R2 e R3 representam as ordenadas dos polígonos que iniciam sua formação na parte inferior da tela à esquerda;
R5 e R6 representam as abscissas dos polígonos que iniciam sua formação na parte superior da tela à esquerda;
R7 e R8 representam as ordenadas dos polígonos que iniciam sua formação na parte superior da tela à esquerda;

A linha 380 traça os polígonos que iniciam na parte inferior da tela à direita.

A linha 390 traça os polígonos que iniciam na parte superior da tela à direita.

A linha 400 traça os polígonos que iniciam à esquerda na parte inferior da tela.

A linha 410 traça os polígonos que iniciam à direita na parte superior da tela.

A linha 420 verifica o número de vezes que o *loop* entre as linhas 170 e 430 foi executado. A cada vez que este *loop* é executado quatro polígonos são traçados (2 na parte inferior e 2 na parte superior da tela). O objetivo deste programa é traçar N polígonos à direita e N polígonos à esquerda, num total de 2N polígonos. A cada *loop* são traçados 4 polígonos; portanto se N for par e I for o número de vezes que o loop deve ser executado teremos:
$I = (2N/4) = N/2$

Porém se N for ímpar, teremos que:
$I = 2(N+1)/4 = (N+1)/2$
ou seja, o *loop* será executado $(N+1)/2$ vezes.

O número de *loops* executados é contato em I. Se o contador I for igual ao número de loops a serem executados pelo programa a próxima instrução a ser executada é a linha 440. Porém, se o contador I for menor do que o número de *loops* a serem executados, o *loop* é executado novamente e o contador I é incrementado de uma unidade.

A linha 450 espera que você pressione qualquer tecla para formar um outro prisma.

A linha 470 faz com que o programa retorne ao início.

Agora que você sabe o que cada linha executa e o que cada variável representa, sugiro que você teste algumas modificações e depois veja os efeitos na tela.

Fig. 6

Digite:

```
50 T1 = 200
100 T6 = 80
```

e verifique a mudança nas coordenadas de referência do traçado dos polígonos, alterando a largura dos polígonos utilizando N = 50.

Digite:

```
70 T3 = 120
110 T7 = 120
```

e verifique a alteração nas coordenadas iniciais, alterando o tamanho dos polígonos para N = 120.

Digite:

```
110 T7 = 80
120 T8 = 150
```

e verifique a alteração nas coordenadas iniciais do polígono da esquerda para N = 50.

Redigite as linhas 50 e 120 da figura 2 e a seguir digite:

```
150 X = 279:Y = 10
```

e verifique para N = 90 a alteração ocorrida com a ordenada de início de traçado dos polígonos superiores.

Digite:

```
385  GOTO 400
405  GOTO 420
```

e verifique que desta forma os polígonos na parte superior da tela não serão traçados.

Retire as linhas 385 e 405 e digite:

```
125  GOTO 150
375  GOTO 390
395  GOTO 410
```

que desta forma os polígonos da parte inferior da tela não serão traçados.

Retire as linhas 125, 375 e 395 e Digite:

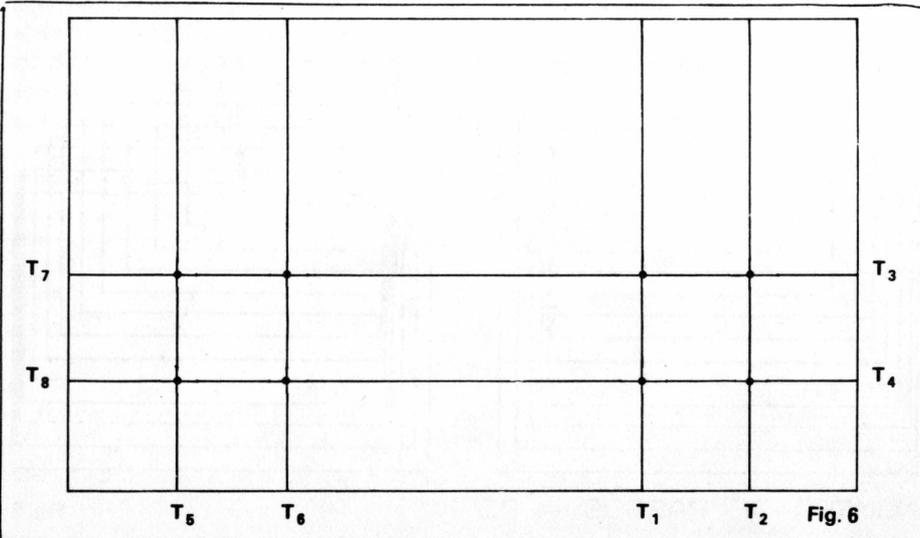
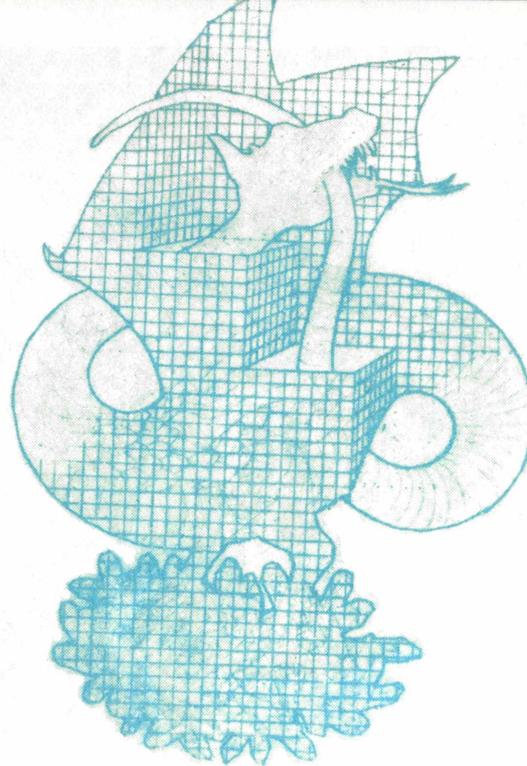
```
135  GOTO 150
385  GOTO 410
```

que serão traçados apenas os polígonos da parte inferior da tela à direita e os polígonos da parte superior à esquerda.

Retire as linhas 135 e 385 e digite:

```
125  GOTO 140
375  GOTO 390
405  GOTO 420
```

que serão traçados os polígonos da parte superior da tela à direita e os polígonos da parte inferior à esquerda.

Agora que você já conhece um pouco mais de gráficos utilizando o comando HPLOT no TK-2000, como foi visto neste programa, você, com suas habilidades, pode desenvolver ótimos programas relacionados com o traçado em alta resolução. Boa sorte!

Memória Ocupada = 2770 bytes
Soma Sintática = 41821
Nível =

DRAGÃO
é tão bonzinho... o Dragão

Programa: Daniel Gerk de Azevedo Quadros
Texto: Nancy Mitie Ariga

Você foi escolhido para ser o rei da Perívia e terá tudo que desejar, pelo resto de sua vida, se provar que merece esta posição. Os habitantes da Perívia desejam uma prova de força (ou sabedoria) e para mostrar a eles a sua capacidade, você passará uma noite na *Ilha do Dragão* levando uma única lança.

Na *Ilha do Dragão* vive um terrível dragão, que passa o seu tempo passeando pelas 12 cavernas (Ø a B) dispostas circularmente na ilha. (Fig. 1)

Digite em qual caverna você supõe que o terrível *dragão* se encontra. Se a lança for atirada na caverna correta, você matará o *dragão* e subirá ao trono ao retornar à Perívia. Caso você não acerte a caverna, vá buscá-la, apertando a tecla *S.*

Mas cuidado! Antes de ir buscar a lança atirada, o *dragão* pode ter tempo de se mover para uma caverna adjacente. Logo, se o *dragão* for para a caverna em que você atirou a lança e você estiver lá, pegando a lança, adivinhe o que acontecerá?. . .

Veja! O tempo está correndo e às oito horas o *dragão* sairá da caverna e o devorará com lança e tudo; portanto apresse-se e use sua massa cefálica para descobrir onde o terrível *dragão* se encontra. Coragem, o povo da Perívia confia em você e espera sua volta triunfal ao amanhecer, para coroá-lo como Rei da Perívia!

Uma pequena informação: as últimas oito pessoas que foram escolhidas para serem coroadas como rei da Perívia e foram à *Ilha do Dragão,* não retornaram. Mas não desanime, você retornará. Para isto, aqui estão algumas dicas para você matar o *dragão* antes das oito:

O *dragão* urra quando a lança é atirada na caverna ao lado dele, e todas as vezes que o *dragão* mudar de caverna, durante a sua caçada, você será informado. Por exemplo, na figura 3, você atirou a sua lança na caverna 8 e o dragão urrou logo, ele pode estar na caverna 7 ou 9, pois ele não se moveu e você pôde tranquilamente buscar sua lança na caverna 8. Já sua figura 2, você atirou a lança na caverna 1 e não acertou porém, após o arremesso da lança, o *dragão* se moveu. Como ele não estava nas cavernas adjacentes à sua, você pode buscar a lança na caverna 1, apertando a tecla *S.*

Digite o programa anexo e aceite o desafio! Mas, antes de digitar o *programa Dragão,* introduza a *Soma Sintática* (Microhobby número 5) na RAM TOP do seu computador para poder, após a digitação do programa, conferir a *Soma Sintática* fornecida no quadro, digitando:

```
PRINT USR 30000
```

se a RAMTOP for o endereço 30000. Lembre-se que você deve primeiramente digitar o programa exatamente como ele está, sem alterar nomes ou números (inclusive as linhas REM), senão o valor da *Soma Sintática* não será o mesmo. Cuidado ao digitar a linha 1040, pois os caracteres são GRAPHIC SPACE e 3 GRAPHIC SHIFT G (lembre-se: a *Soma Sintática* exige que todos os caracteres sejam digitados exatamente como a listagem é apresentada).

Para saber qual a memória ocupada apenas pelo programa, digite:

```
PRINT PEEK 16396+256*PEEK 16397
-16509
```

que ele fornecerá o número de bytes que o programa ocupa na memória do seu computador.

Digite GOTO 4000 para gravar o programa. Assim, quando você terminar de carregar o programa ele imediatamente começa a rodar (AUTO — START).

DRAGÃO
5:00
CAVERNA?
DRAGÕES JANTAM ÀS 8:00

DRAGÃO
5:00
CAVERNA?1
ERROU...
MOVEU
BUSCA?S
DRAGÕES JANTAM ÀS 8:00

DRAGÃO
5:45
CAVERNA?8
URRO
DRAGÕES JANTAM ÀS 8:00

O PROGRAMA

```
   1 REM DANIEL G. A. QUADROS
 120 GOSUB 1000
 125 RAND
 130 LET H=5
 135 LET M=0
 140 PRINT AT 21,0;"DRAGOES JANT
AM AS 8:00"
 150 LET D=INT (12*RND)
 160 GOSUB 3000
 165 PRINT AT 6,23;"CAVERNA? "
 166 PRINT AT 7,23;"         "
 167 PRINT AT 8,23;"         "
 168 PRINT AT 9,23;"         "
 170 LET C$=INKEY$
 180 IF C$="" THEN GOTO 170
 190 IF CODE C$<28 OR CODE C$>39
 THEN GOTO 170
 195 PRINT AT 6,31;C$
 200 LET C=CODE C$-28
 210 LET X=2*VAL P$(2*C+1)
 220 LET Y=2*VAL P$(2*C+2)
 230 FOR I=1 TO 5
 240 PRINT AT Y,X;"■"
 250 GOSUB 3200
 260 PRINT AT Y,X;CHR$ (C+28)
 270 GOSUB 3200
 280 NEXT I
 290 IF D=C THEN GOTO 900
 300 IF ABS (D-C)=1 OR ABS (C-D)
=11 THEN GOSUB 3300
 310 PRINT AT 7,23;"ERROU..."
 320 LET S=INT (3*RND-1)
 330 IF S=0 THEN GOTO 380
 340 LET D=D+S
 350 IF D<0 THEN LET D=12+D
 360 IF D>11 THEN LET D=D-12
 370 PRINT AT 8,23;"MOVEU"
 380 PRINT AT 9,23;"BUSCA? "
 390 LET R$=INKEY$
 400 IF R$="" THEN GOTO 390
 405 PRINT AT 9,29;R$
 410 IF R$="N" THEN GOTO 450
 420 IF R$<>"S" THEN GOTO 390
 430 IF C=D THEN GOTO 800
 450 LET M=M+15
 460 IF M=60 THEN LET H=H+1
 470 IF M=60 THEN LET M=0
 475 GOSUB 3000
 480 IF H=8 THEN GOTO 800
 490 IF R$="S" THEN GOTO 160
 500 PRINT AT 8,23;"        "
 510 GOTO 320
 800 REM PERDEU
 810 FOR I=1 TO 5
 820 PRINT AT 21,0;"O DRAGAO ENG
ORDA...VOCE PERDEU."
 830 GOSUB 3200
 840 PRINT AT 21,0;"O DRAGAO ENG
ORDA...VOCE PERDEU."
 850 GOSUB 3200
 860 NEXT I
 870 GOTO 970
 900 REM ACERTOU
 910 FOR I=1 TO 5
 920 PRINT AT 21,0;"PARABENS...V
OCE ACERTOU."
 930 GOSUB 3200
 940 PRINT AT 21,0;"PARABENS...V
OCE ACERTOU."
 950 GOSUB 3200
 960 NEXT I
 970 PRINT AT 21,0;"VAMOS JOGAR
DE NOVO?                   "
 975 LET R$=INKEY$
 980 IF R$="N" THEN STOP
 985 IF R$="S" THEN GOTO 130
 990 GOTO 975
1000 REM DESENHA CAVERNAS
1005 CLS
1010 FOR I=5 TO 15
1020 PRINT AT 0,I;"■"
1030 PRINT AT 20,I;"■"
1040 PRINT AT I,0;"■■■■"
1050 PRINT AT I,17;"■■■■"
1060 NEXT I
1070 FOR I=0 TO 20
1080 PRINT AT 4,I;"■"
1090 PRINT AT 16,I;"■"
1100 PRINT AT I,4;"■"
1110 PRINT AT I,16;"■"
1120 NEXT I
1130 FOR I=4 TO 12 STEP 4
1140 FOR J=1 TO 3
1150 PRINT AT I,J;"■"
1160 PRINT AT I,16+J;"■"
1170 PRINT AT J,I;"■■■"
1180 PRINT AT 16+J,I;"■■■"
1190 NEXT J
1200 NEXT I
1210 PRINT AT 8,10;"■"
1220 PRINT AT 9,9;"■■■"
1230 PRINT AT 10,9;"■ ■"
1240 PRINT AT 11,9;"■ ■"
1250 LET P$="315171939597795939171513"
1260 FOR I=0 TO 11
1270 LET X=2*VAL P$(I*2+1)
1280 LET Y=2*VAL P$(I*2+2)
1290 PRINT AT Y,X;CHR$ (28+I)
1300 NEXT I
1310 PRINT AT 1,24;" DRAGAO "
1320 RETURN
3000 REM RELOGIO
3010 PRINT AT 3,26;CHR$ (28+H);"
:";
3020 PRINT CHR$ INT (M/10+28);
3030 PRINT CHR$ (M-10*INT (M/10)
+28)
3040 RETURN
3200 REM PAUSA
3210 RETURN
3300 REM URRO
3310 FOR I=1 TO 3
3320 PRINT AT 13,25;" URRO "
3330 GOSUB 3200
3340 PRINT AT 13,25;" URRO "
3350 GOSUB 3200
3360 NEXT I
3370 PRINT AT 13,26;"      "
3380 RETURN
4000 SAVE "DRAGAO"
4010 GOTO 100
```

CURSO DE   TK

aula 11

Sub-rotinas

Piazzi — Rossini

Quando uma conseqüência de instruções deve ser repetida diversas vezes num programa, ela pode ser escrita apenas uma vez como se fôsse um pequeno programa reparado. Este pequeno programa deve ser "acessado" ou "chamado" pela key-word GOSUB (tecla H) e deve obrigatoriamente terminar com uma key-word, que faz com que o programa volte a linha logo após o GOSUB; esta key-word é RETURN (tecla Y). A esta seqüência de instruções chamamos *sub-rotina.* Repare no seguinte programa:

```
 10 SLOW
 20 LET C=1
 30 GOSUB 280
 40 FOR P=1 TO 11
 50 PRINT AT 12-P,P+5;CHR$ X
 60 NEXT P
 70 FOR N=1 TO 11 STEP 2
 80 GOSUB 280
 90 FOR P=1 TO 12-N
100 PRINT AT -1+P+N,15+P;CHR$ X
110 NEXT P
120 GOSUB 280
130 FOR P=1 TO 12-N
140 PRINT AT 10+P,28-P-N;CHR$ X
150 NEXT P
160 GOSUB 280
170 FOR P=1 TO 11-N
180 PRINT AT 23-P-N,17-P;CHR$ X
190 NEXT P
200 GOSUB 280
210 FOR P=1 TO 11-N
220 PRINT AT 13-P,5+P+N;CHR$ X
230 NEXT P
240 NEXT N
250 LET C=C*(-1)
260 LET X=0
270 GOTO 30
280 IF C=1 THEN LET X=1+INT (12
7*RND)
290 IF X>63 THEN LET X=X+64
300 RETURN
```

Note que nas linhas 3Ø, 8Ø, 12Ø, 16Ø e 2ØØ "chamamos" a mesma seqüência de instruções (sub-rotina) que estão nas linhas 28Ø e 29Ø, terminando com um RETURN. Assim, ao chegar na linha 3Ø, o TK passa às linhas 28Ø, 29Ø e 3ØØ e volta para a linha 4Ø; ao chegar na linha 8Ø, ele vai novamente para as linhas 28Ø, 29Ø e 3ØØ mas volta agora para a linha 9Ø. E assim por diante!

As sub-rotinas têm um símbolo especial para uso em diagramas de bloco; veremos isto logo mais nesta mesma aula.

Zapplenum — A Key-Word REM

À medida que os programas se tornam mais complicados, seria útil que pudessemos colocar alguns "comentários" junto à listagem para esclarecer melhor os detalhes. Além disso, seria bom se pudessemos dar nomes aos programas. Para isto temos a palavra chave *REM* (tecla E) que não executa nada; apenas permite a inserção de comentários nos programas.

Faremos agora um pequeno "jogo" que demonstra a utilização de muitos comandos e estruturas do BASIC, inclusive um truque novo com relação ao INKEY$'; vejamos:

```
   1 REM PIAZZI-ROSSINI
  10 REM ZAPPLENUM
  20 SLOW
  30 RAND
  40 PRINT "QUANTOS NUMEROS ?"
  50 INPUT N
  60 PRINT "TEMPO EM ZUTUS ?"
  70 INPUT T
  60 PRINT "DIFICULDADE (0,1,2)?
"
  90 INPUT D
 100 CLS
 104 LET X=10
 106 LET Y=X
 110 LET S0=0
 115 LET S=S0
 120 PRINT "SCORE ";S;TAB 12;"ZA
PPLENUM"
 125 REM INICIO DO JOGO
 130 FOR I=1 TO N
 140 LET A=INT (21*RND)+1
 150 LET B=INT (31*RND)
 160 PRINT AT A,B;I
 170 FOR J=1 TO T-D*I
 180 PRINT AT Y,X;"■"
 190 PRINT AT Y,X;" "
 200 LET X=X+(INKEY$="8")-(INKEY
$="5")
 210 LET Y=Y-(INKEY$="7")+(INKEY
$="6")
 220 IF X<0 THEN LET X=31
 230 IF X>31 THEN LET X=1
 240 IF Y<0 THEN LET Y=21
 250 IF Y>21 THEN LET Y=1
 260 PRINT AT 0,12;"BEATLES     "
 270 IF X=B AND Y=A THEN GOSUB 3
50
 280 PRINT AT 0,12;"BEATLES     "
 290 NEXT J
 300 NEXT I
 310 CLS
 320 PRINT AT 10,10;"FIM DO JOGO
"
 330 IF S>50 THEN PRINT "MUITO B
EM..."
 340 STOP
 350 LET S=S+I
 360 PRINT AT 0,6;S
 370 FOR K=1 TO 15
 380 PRINT AT 0,24;"NHOC..."
 390 PRINT AT 0,24;"NHOC..."
 400 NEXT K
 410 PRINT AT 0,24;"       "
 420 LET J=T-D*I-2
 430 RETURN
```

O TK gera números em posições aleatórias da tela que devem ser "comidos" por você (■) antes que apareça algum novo número; assim, por exemplo, se você estiver perseguindo o número 4 e, antes de alcançá-lo, aparece o número 5 na tela, não adianta mais "comer" o número 4 pois seu SCORE não será alterado. Quanto maior for o número, menor o tempo que você terá para "comê-lo", em contrapartida, mais você ganha se conseguir! Se for gerado um número com dois algarismos, você deve "comer" o mais significativo (Por quê?).

Brinque então um pouco com ele usando para começar, dez números, tempo 18 ZUTUS e nível 1 de dificuldade. O tempo é contato numa unidade de qualquer (ZUTUS = Zapple Universal Time Unity) pois a relação com as unidades convencionais não é trivial. Neste programa é interessante salientar as linhas 200 e 210: quando aparece mais do que uma igualdade (e/ou desigualdade) numa expressão matemática, apenas a primeira, contando da esquerda para a direita, é considerada como tal. As demais, são consideradas como parte de uma afirmação lógica que pode ser verdadeira ou falsa: se verdadeira, produz resultado 1 e, se falsa, produz resultado zero. Assim, na linha 200, se no instante em que o TK estiver executando a instrução, a tecla 8 estiver pressionada (⇩), a primeira afirmação é verdadeira e a segunda é falsa; assim teremos:

```
200 LET X=X+1-0
```

Note que este tipo de estrutura, permite o uso das operações lógicas AND, OR e NOT; assim, por exemplo, seria válido escrever:

```
LET H=H+(INKEY$<>"8" AND I=5)
```

ou

```
LET P=P-5*(A<=3 OR K=A+B)
```

Para entender melhor, procure fazer alguns comandos diretos do tipo:

```
PRINT (6>10)
PRINT (3=3)
PRINT (NOT (2<1))
PRINT NOT 0
PRINT (9=9 AND 7>5)
PRINT (3<4 OR 4<3)
```

Quanto ao resto do programa, acreditamos que você possa entendê-lo sem dificuldades se dedicar um pouquinho de tempo sobre sua lógica. Note os detalhes, como o "pisca-pisca" das palavras BEATLES e NHOC . . . Note também que a sub-rotina utilizada não precisaria necessariamente ser uma sub-rotina; bastaria fazer GOTO 350 ao invés de GOSUB (na linha 270) e GOTO 280 ao invés de RETURN (na linha 430). Para facilitar seu raciocínio, iremos apresentar o fluxograma do programa.

C
I = 1 TO N
A = INT (21*RND) + 1
B = INT (31*RND)
AT A, B, I
A
J = 1 TO T-D*J
AT x,y "■"
AT x,y " "
x = x+(INKEY$ = "8") – (INKEY$ = "5")
y = y+(INKEY$ = "7") – (INKEY$ – "6")
B

B
x < 1 ? SIM x = 21 NÃO
x > 21 ? SIM x = 1 NÃO
y < 0 ? SIM y = 31 NÃO
y > 31 ? SIM y = 1 NÃO
AT 0,12 "BEATLES"
x = A AND y = B SIM SCORE S,J,T,D,I NÃO
AT 012 "BEATLES"
A
CLS
"GAME OVER"
S > S0 ? "MUITO BEM"
F

Note o símbolo utilizado para representar "chamadas" de sub-rotinas

e

para iniciar sub-rotinas . Costuma-se associar um *nome* à sub-rotina o qual é colocado na parte superior do símbolo e, na parte inferior, colocam-se as variáveis do programa principal que serão utilizadas pela sub-rotina. Você saberia explicar para que serve a instrução da linha 420?

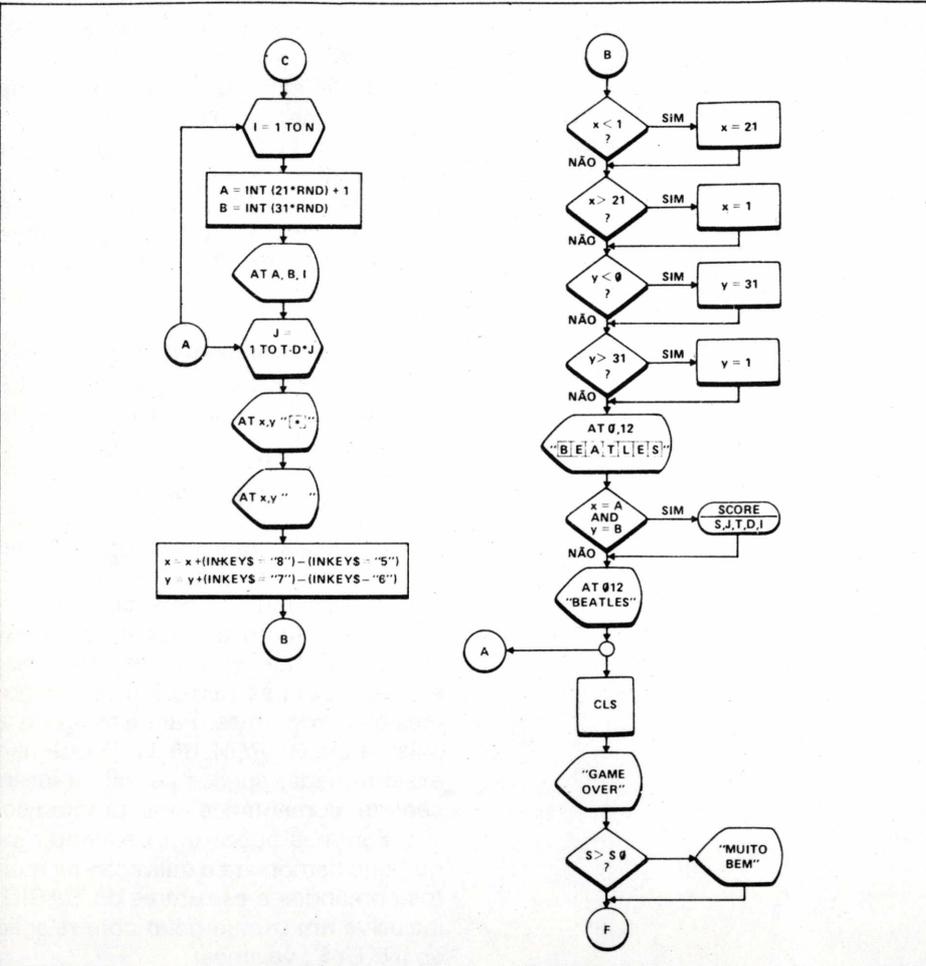
EXERCÍCIOS

1. Faça um programa capaz de ler dois números inteiros e calcular seu M.D.C. (Máximo Divisor Comum) e M.M.C. (Mínimo Múltiplo Comum).

2. Elabore um programa capaz de "ler" duas matrizes e, se possível, dependendo da compatibilidade de dimensões, calcule o produto das mesmas.

3. Faça um programa que gere na tela a série de *Fibonacci:*

0 1 1 2 3 5 8 13 21 . . . . .

4. Tente simular a ***rotação*** de um objeto na tela. ○

RESPOSTAS DO CURSO DE BASIC

aula 10

EXERC 1

```
 75 PLOT I,21
```

EXER. 2 (A E B)

```
  10 FAST
  20 FOR X=0 TO 63
  30 LET Y=ABS (40*SIN X)
  40 PLOT X,Y
  50 PLOT X,0
  60 NEXT X
```

```
   5 PRINT "INTRODUZA ""D"""
   6 INPUT D
   7 CLS
  10 FAST
  20 FOR X=0 TO 63
  30 LET Y=(EXP (-D*X))*ABS (40*
SIN X)
  40 PLOT X,Y
  50 PLOT X,0
  60 NEXT X
```

EXERC. 3

```
  10 PRINT "ENTRE COM ANGULO EM
GRAUS"
  20 INPUT G
  30 LET R=(PI*G)/180
  40 PRINT
  50 PRINT G;" GRAUS= ";R;" RADI
ANOS"
  60 PRINT
  70 PRINT "ENTRE COM A FUNCAO D
ESEJADA (DI-GITE SHIFT FUNCTION
E FUNCAO MA-TEMATICA)"
  80 INPUT C$
  90 LET T$=C$+STR$ G
 100 LET C$=C$+STR$ R
 110 PRINT
 120 PRINT T$;"=";VAL C$
```

EXERC. 4

```
  10 PRINT "INTRODUZA UM VALOR"
  20 PRINT
  30 INPUT V
  40 LET X=(LN V)/LN 10
  50 PRINT "SEU LOGARITMO NA BAS
E 10 E■:"
  60 PRINT
  70 PRINT "LOG   ";V;" = ";X
  80 PRINT "    10"
```

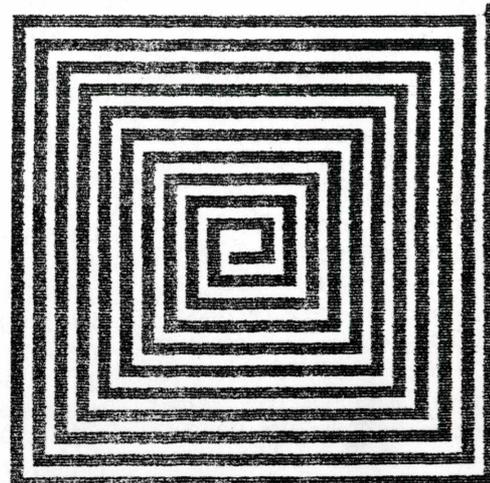
EXERC. 5

```
  10 PRINT "VAMOS CALCULAR A DIS
TANCIA ENTREO NAVIO E O FAROL."
  20 PRINT "ENTRE COM A ALTURA D
O FAROL: ";
  30 INPUT H
  40 PRINT H
  50 PRINT
  60 PRINT "AGORA ENTRE COM O AN
GULO QUE A  LUZ EMITIDA FORMA CO
M O   FAROL   (EM GRAUS)";
  70 INPUT A
  80 LET A=(PI*A)/180
  90 LET D=(TAN A)*H
 100 PRINT ,,,,"DISTANCIA = ";D
```

EXERC. 6

```
   5 SLOW
  10 LET M=21
  20 LET N=31
  30 LET Z=2
  40 FOR N=N TO N+Z STEP SGN Z
  50 PLOT N,M
  60 NEXT N
  70 FOR M=M TO M+Z STEP SGN Z
  80 PLOT N,M
  90 NEXT M
  95 LET Z=Z+SGN Z*2
 100 LET Z=-Z
 110 GOTO 40
```

por dentro do apple

Prof. Wilson José Tucci – Coordenador de Projetos Especiais da Escola Experimental Pueri Domus.

Parte I

Você acredita em discos voadores? Bem, quer você acredite ou não, poderá se divertir com eles nesse jogo simples e interessante, que introduz mais algumas técnicas úteis para seus próprios jogos.

Neste jogo, feito com os gráficos de baixa resolução do Apple, você tem uma base de canhões e tem como objetivo, destruir os invasores marcianos (ou de onde você quiser que eles sejam).

Consideremos os problemas envolvidos num jogo desse tipo; o algoritmo geral para tal jogo pode ser assim escrito:

1. Inicializar o placar, que vai marcar quantos discos já passaram, quantos você acertou e outras informações.
2. Desenhar o canhão.
3. Desenhar o disco (de preferência com uma cor diferente da do canhão).
4. Ver se o jogador atirou.
5. Se atirou, então devemos:

5.1. avançar o tiro e o disco.

5.2. verificar se o tiro acertou o disco.

Se acertou, devemos fazer o disco cair. Se não acertou, voltamos para 5.1 enquanto o disco não fugir.

6. Se não atirou, devemos avançar o disco e voltar ao passo 4 enquanto o disco não fugir.
7. Atualizar o placar e voltar ao passo 2.

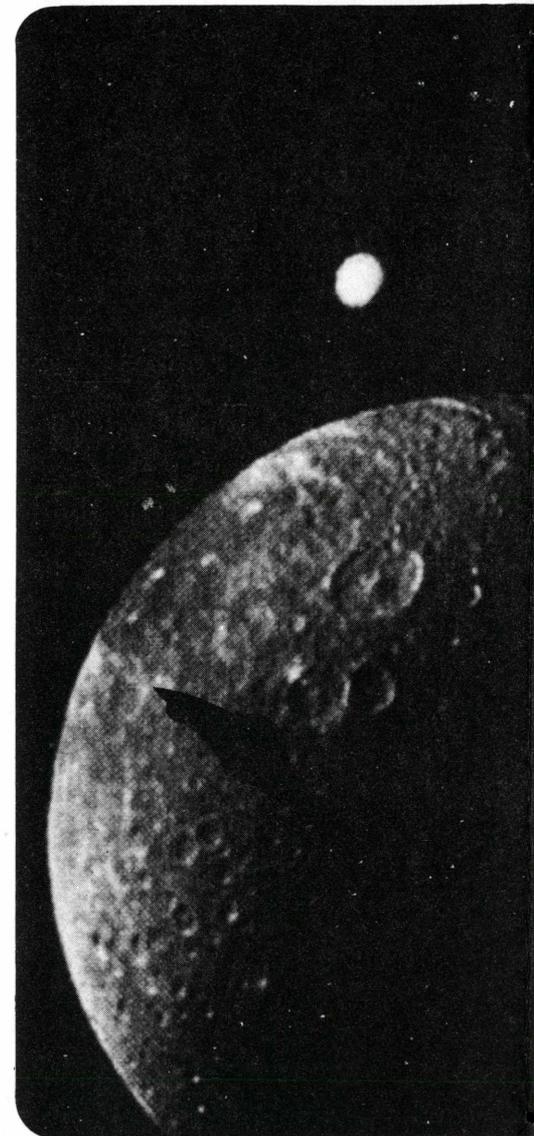
Além da animação visual, um jogo deve ter também animação musical (barulho). Todo barulho no Apple é feito referenciando-se à posição de memória – 16336 (hexadecimal CØ3Ø para aqueles com 16 dedos). Essa referência se faz com um *PEEK (– 16336)*, a partir do Applesoft, ou com uma *LDA $ CØ3Ø* ou *STA $CØ3Ø* a partir de Linguagem de Máquina. Tais comandos produzem um "click" no alto falante do Apple, e todos os sons gerados são seqüências de "clicks". Existem pelo menos três maneiras de se fazer barulho no Apple:

1. Fazer em BASIC comandos que executem vários *PEEK (– 16336)*.
2. Executar um *PRINT CHR$ (7)*, ou seja *CTRL-G*, que automaticamente, executa uma série de referências à posição – 16336.
3. Fazer uma sub-rotina em Linguagem de Máquina que executa uma série de *LDA $CØ3Ø* ou *STA $CØ3Ø*, e chamar essa sub-rotina através do comando BASIC, *CALL adr*, onde *adr* é a posição da memória onde começa a sub-rotina.

O programa

Linha 1ØØ: executa a sub-rotina que carrega o programa em Linguagem de Máquina que faz o barulho. A seguinte listagem é do programa carregado, a partir da posição 771 (hexadecimal 3Ø3).

```
0303-   AD 30 C0    LDA   $C030
0306-   88          DEY
0307-   D0 04       BNE   $030D
0309-   C6 07       DEC   $07
030B-   F0 08       BEQ   $0315
030D-   CA          DEX
030E-   D0 F6       BNE   $0306
0310-   A6 06       LDX   $06
0312-   4C 03 03    JMP   $0303
0315-   60          RTS
```

É comum colocar rotinas para carregar programas em Linguagem de Máquina diretamente em BASIC, isso evita que o usuário tenha que digitar dois programas diferentes, um em Linguagem de Máquina e outro em BASIC. É fácil perceber como isto é feito, analisando as linhas 98Ø e 99Ø: um *FOR NEXT; "POKEia"* os valores decimais dos bytes do programa em Linguagem de Máquina. Se você quiser ver o programa como na listagem acima, basta fazer *CALL – 151* (isso chama o monitor) e depois, quando aparecer o asterisco, digite *3Ø3L,* isso vai listar o programa em Linguagem de Máquina armazenado a partir do byte 3Ø3 (hexadecimal). Após a listagem do programa, vão aparecer mais alguns comandos, usualmente BRK. Eles não fazem parte do programa, estão lá por que é assim que o APPLE "acorda". Em todo caso, não serão executados, pois o programa acaba no byte 315, *RTS,* que equivale ao comando BASIC *RETURN* (lembre-se que *CALL* é uma chamada de sub-rotina, como *GOSUB*).

Linha 12Ø: Q é o número da posição da memória que lê o teclado; o valor des-

Disco Voador

O PROGRAMA

```
0  REM @POR DENTRO DO APPLE

         DISCO VOADOR
100  GOSUB 980
110  HOME : INVERSE
120 Q =  - 16384:BU =  - 16336:G$
     =  CHR$ (7)
130  GR
140  VTAB 23: HTAB 1: PRINT ":DIS
    COS::";: HTAB 11: PRINT "::T
    IROS::";: HTAB 21: PRINT ":A
    CERTOS:";: HTAB 31: PRINT ":
    :::%::::"
150  VTAB 24: HTAB 1: PRINT "::::
    :::::";: HTAB 11: PRINT ":::
    ::::::";: HTAB 21: PRINT "::
    :::::::";: HTAB 31: PRINT ":
    ::00%:::";
160  GOSUB 580:SL = 35:XF = 1:YY =
    0:HR = 0:CD =  INT ( RND (1)
     * 14) + 1: GOSUB 940
170  VTAB 21: HTAB 1
180  PRINT "<ESPACO> ATIRA, OUTRA
     PARA A BALA"
190  GOSUB 940
200  REM ===CANHAO===
210 CC =  INT ( RND (1) * 14) + 1
    : IF CC = CD THEN 210
220  COLOR= CC: HLIN 22,26 AT 39:
     HLIN 23,25 AT 38: VLIN 36,3
    7 AT 24
580 AD = AD + 1: GOSUB 900
590 HR = 0: RETURN
940 YF = Y1:Y1 =  INT ( RND (1) *
    20) + 10: RETURN
950  POKE 6,100 - (3 * N): POKE 7
    ,5
960  CALL 771
970  RETURN
980  FOR I = 771 TO 789: READ A: POKE
    I,A: NEXT : RETURN
990  DATA  173,48,192,136,208,4,1
    98,7,240,8,202,208,246,166,6
    ,76,3,3,96
```

se byte é o código ASCII da tecla pressionada (serve para ler o teclado sem aquele cursor do *GET, BU,* como já discutido, é a posição que faz tocar o alto-falante, e *G$* é o sininho.

Linhas 140 e 150: preparam o placar nas últimas duas linhas da página de texto (as linhas 1 a 20 estão ocupadas pelo gráfico de baixa resolução).

Linha 160: completa o placar (através da sub-rotina 580), inicializa uma série de parâmetros para o disco, incluindo uma cor aleatória (CD), e a posição vertical do disco, também aleatória, através da sub-rotina 940.

Linhas 170 e 180: mostram a única instrução importante desse jogo. Para atirar uma bala você deve apertar o «espaço», e só pode atirar uma bala por disco. Contudo, após atirar, você pode parar a bala apertando qualquer outra tecla e, para fazer a bala voltar a andar, basta apertar novamente qualquer tecla.

Linhas 210 e 220: desenham o canhão, com uma cor aleatória. Repare que a comparação na linha 210 garante que a cor do canhão será diferente da cor do disco.

○

CURSO DE ASSEMBLY

aula 10

Flavio Rossini

Os Saltos Condicionais Baseados nos FLAGS

As instruções, JP e JR, como foram usadas na aula passada, são chamadas de saltos INCONDICIONAIS. Assim, como no BASIC a instrução GOTO ganha bastante poder, se usada juntamente com a instrução IF/THEN; temos em Linguagem de Máquina algo equivalente que são os saltos CONDICIONAIS, ou seja, "Salte se ocorrer dada condição".

Há quatro condições que podem ser testadas usando JR e 8 usando JP. Duas delas se baseiam no CARRY: salte se o CARRY for 1 e salte se o CARRY for zero:

> JR C dado ('38' + 1 byte para dado)
> (salte relativo se CARRY = 1)
> JR NC dado ('30' + 1 byte para dado)
> (salte relativo se CARRY = 0 NC = NO CARRY)
> JP C endereço ('DA' + 2 bytes para endereço)
> JP NC endereço ('D2' + 2 bytes para endereço)

Para poder entender as outras condições, vamos antes estudar um pouco mais o registro dos flags (F) que, como sabemos, tem 8 bits, sendo um deles para indicar o flag de CARRY. Vamos agora ver mais alguns FLAGS.

Temos um flag de paridade (flag P) que assinala se a quantidade de uns ou zeros do ACUMULADOR é par (EVEN) ou ímpar (ODD) *após efetuada uma operação aritmética ou lógica;* assim temos:

> ***JP PE*** endereço ('EA' + 2 bytes para endereço) (JUMP if PARITY is EVEN) ***JP PO*** endereço ('32' + 2 bytes para endereço) (JUMP if PARITY is ODD)

Além disto, *este mesmo flag* serve para testar se houve OVERFLOW (flag 0) que seria espécie de CARRY para quando trabalhamos com números positivos e negativos. Entretanto, ela é apenas um flag de ALARME, não permitindo identificar qual o resultado real (como era possível com o CARRY usando ADC ou SBC). De fato, quando calculamos usando a convenção de números negativos, podem haver "trocas" acidentais de sinais; por exemplo, se você somar '4A' que é positivo com '44' que também é positivo, você obtém '8E' que é negativo, pois agora o maior número positivo representável em 1 byte é 127 ('7F'). Note que você terá OVERFLOW mas não CARRY = 1 pois, usando a convenção de números positivos esta soma produz CARRY = 0. Desta forma este FLAG funciona de maneira que JP PE equivale a JUMP se houve OVERFLOW e JP PO equivale a JUMP se não houver OVERFLOW. Não existe salto relativo usando este flag como condição. Note que por ser um flag DUPLO, algumas instruções afetam o flag como sendo P (paridade) e, outras como O (OVERFLOW).

Outro flag é o flag Z que indica se o *resultado de uma operação* é zero; assim temos:

> JR Z dado ('28' + 1 byte para dado)
> ***(jump relative if zero)***
> JR NZ dado ('20' + 1 byte para dado)
> ***(jump relative if not zero)***
> JP Z endereço ('CA' + 2 bytes para endereço)
> JP NZ endereço ('C2' + 2 bytes para endereço)

O último flag que iremos estudar nesta altura do curso é o flag de sinal (**S**) que também só tem sentido em ser usado quando utilizamos a convenção de números negativos; desta forma, você pode saltar dependendo se o *resultado de uma operação* é negativo ou positivo:

JP M endereço ('FA' + 2 bytes para endereço)
(jump if minus)
JP P endereço ('F2' + 2 bytes para endereço)
(jump if plus)

Também não existe salto relativo para esta condição.

Obviamente as instruções de JUMP não afetam nenhum FLAG; mas note que agora não podemos nos preocupar somente com o flag de CARRY. Para cada instrução, devemos saber PRECISAMENTE quais flags são afetados; por exemplo, as instruções de LD não afetam nenhum flag e as instruções de ADD e SUB (ADC e SBC também afetam TODOS os flags. Procuraremos, na medida do possível, explicitar os flags afetados sempre que falamos em alguma instrução. Note que este curso é um resumo do meu livro: Introdução à Linguagem de Máquina para o TK — Assembly Z-8Ø " vol I ; nele você poderá encontrar no apêndice II, todas as instruções e o efeito de cada uma sobre FLAGS. Além disso, vários exercícios são apresentados.

Apenas para complementar as instruções com números negativos, vamos apresentar a instrução *NEG* que "muda o sinal" do número que está no acumulador; seu código é 'ED44'.

Como você deve ter notado anteriormente nas instruções de salto condicional, nós sempre sublinhamos a frase: *RESULTADO DE UMA OPERAÇÃO.* De fato, os JUMPs condicionais funcionam se antes houve operação que AFETOU os flags *adequadamente.* Por exemplo, NÃO adianta carregar o acumulador com o conteúdo de algum registro ou memória e, em seguida querer usar um JUMP condicional baseado no conteúdo do acumulador! Isto simplesmente utilizará para testes, os FLAGS da ÚLTIMA OPERAÇÃO que foi EFETUADA antes deste carregamento. Logo, se você quer "saltar" para algum ponto (3ØØ2Ø), baseado por exemplo no fato de que uma determinada memória (3ØØ1Ø) contenha o número zero, NÃO adianta fazer o seguinte:

LD A,3ØØ1Ø (não afeta nenhum flag)
JP Z,3Ø12Ø (testa o flag "gerado" na última operação)

Isto é válido para todos os saltos condicionais. Para poder testar condições nesses casos é necessária então, uma nova instrução chamada CP (COMPARE) que compara o conteúdo do ACUMULADOR com qualquer outro registro, com um dado número (entre Ø e 255 ou entre — 128 e 127) ou memória endereçada por HL; ela funciona da seguinte maneira: *sem alterar* nenhum registro, ela verifica quanto SERIA o resultado do ACUMULADOR MENOS o registro ou dado comparado, atualizando os flags convenientemente; assim, se o número comparado por *igual* ao computador, o flag de *zero* é setado (colocado em 1), se for menor do que o acumulador o flag de sinal indicará *positivo* (Ø) e, se for maior, *negativo* (1). Isto aliado às instruções JP ou JR equivale em BASIC a fazer:

$$\text{IF} \begin{Bmatrix} A = B \\ A > B \\ A < B \end{Bmatrix} \text{THEN GOTO XXXX.}$$

Repare que, para testar se o acumulador é "maior ou igual" ou "menor ou igual" *dois* flags devem ser testados!

Assim, para que o exemplo anterior funcionasse, deveríamos fazer o que mostramos na tabela I.

Obviamente, a instrução CP afeta TODOS os flags; veja os códigos na tabela II.

Completando o ANTI-SCROLL: LOOPS em linguagem de máquina

Vamos a título de exemplo, ou seja, vamos fazer em Linguagem de Máquina o que a instrução de PRINT 32 brancos fazia: para isto é necessário notar que as instruções de DEC *afetam* o flag de zero (Z) se ao decrementar dado registro ou memória, o resultado der zero. Isto é indicado no flag Z! Acompanhe na tabela III.

Como você pode verificar, este procedimento coloca o número zero, que é o código de "espaço em branco" para TK, nas memórias que correspondem à primeira linha da tela; após o LDDR o par HL está novamente indicando o começo da tela; basta somar 32 para acharmos o endereço do último caractere da primeira linha; ao carregar DE com 32 teremos D = 'ØØ' e E = '2Ø' ('2Ø' = 32) e poderemos então usar o registro E como "contador" para colocar os 32 "brancos" necessários na memória! Ao colocar o último "branco", DEC E fará E = 'ØØ' colocando o flag Z em 1 e o programa não mais saltará para o endereço 3ØØ22, voltando então para o BASIC. Note que o par BC continua 'ØØØØ'. Assim, como você pode notar, fizemos um *loop com Linguagem de Máquina* utilizando a instrução DEC e usando o salto condicional baseado no FLAG de zero, gerado!

Vamos testar o programa:

```
1005 SLOW
1010 FOR I=1 TO 80
1020 RAND USR 30000
1030 PRINT AT 0,0;TAB I; I
1040 NEXT I
```

(Obs.: teste a sub-rotina em Linguagem de Máquina usando JP NZ 3ØØ22 e, depois, JR NZ — 6)

Note que não nos interessa a "saída" da sub-rotina de Linguagem de Máquina (ou seja, par BC), mas precisamos chamá-la para que ela seja executada; utilizamos então a instrução RAND que, no caso, não faz nada, apenas permite que o cursor mude para F *após uma Key-word* para colocar USR pois, sem isso, a linha não é aceita no programa BASIC.

Existe em Linguagem de Máquina, uma instrução especial que facilita executar LOOPS:

DJNZ dado '1Ø' + 1 byte para o dado

Ela usa o registro B como contador e executa um *salto relativo* enquanto que B não atinge zero:

Assim, o final do programa anterior poderia ser como mostra a tabela IV.

Teste então esta terceira possibilidade.

Para finalizar os saltos condicionais, vamos aprender um "truque" que nos permite usar JR em vez de JP, quando queremos comparar um número com o acumulador e saber se o mesmo é maior ou menor: se o número a ser comparado for MAIOR, a diferença do acumulador (supondo-o positivo) menos o número resulta negativa e, portanto, gera CARRY = 1. Logo, apesar de *não* existir a instrução JR M você pode usar JR C. Obviamente, a instrução, *inexistente,* JR P, pode ser simulada por JR NC. Naturalmente isto só funciona se adotarmos a convenção de números positivos ou, se adotarmos a outra, garantirmos que no acumulador esteja um número positivo. Perceba que a instrução JR NC considera Ø como número positivo (o mesmo é válido para JP NC ou JP P.)

O Stack Pointer: Sub-rotinas

Mesmo em Linguagem de Máquina podemos ter sub-rotinas. Temos então uma instrução equivalente ao GOSUB que seria: CALL-endereço ('CD' + 2 bytes para endereço); para retornar ao programa principal, temos a já famosa RET ('C9'). Assim como em BASIC seria possível uma estrutura do tipo JF condição THEN GOSUB XX

LD	**A, 3ØØ1Ø**		**; Coloque em A o conteúdo da memória 3ØØ1Ø**
CP	**Ø**	**'FEØØ'**	**; Compare A com o número Ø; se for igual, flag Z = 1, caso contrário, flag Z = Ø**
ZP	**Z 3Ø12Ø**		**; pule papa 3Ø12Ø se o flag Z = 1, caso contrário, prossiga (JP M seria usado para testar se A 0 e JP P para testar se A Ø**

Tabela I

INSTRUÇÃO			**CÓDIGO**
CP	**dado**	**'FE'**	**+ 1 byte para o dado**
CP	(HL)	'BE'	(você vê alguma utilidade para esta introdução?)
CP	A	'BF'	
CP	B	'B8'	
CP	C	'B9'	
CP	D	'BA'	
CP	E	'BB'	
CP	H	'BC'	
CP	L	'BD'	

Tabela II

3ØØØØ	**LD**	**BC, 726**	**'Ø1D6Ø2'**	
3ØØØ3	**LD**	**HL,(16396)**	**'2AØ64Ø'**	
3ØØØ6	**ADD**	**HL,BC**	**'Ø9'**	
3ØØØ7	**LD**	**D,H**	**'54'**	
3ØØØ8	**LD**	**E,L**	**'5D'**	
3ØØØ9	**LD**	**BC, 693**	**'Ø1B5Ø2'**	
3ØØ12	**LD**	**HL,(16396)**	**'2AØC4Ø'**	
3ØØ15	**ADD**	**HL,BC**	**'Ø9'**	
3ØØ16	**LDDR**		**'EDB8'**	
3ØØ18	**LD**	**DE,32**	**'112ØØØ'**	**; carrega DE com 32**
3ØØ21	**AD**	**HL,DE**	**'19'**	**; coloca em HL o endereço da última posição da primeira linha da tela**
3ØØ22	**LD**	**(HL),Ø**	**'36ØØ'**	**; coloca "espaços em branco" (código Ø) na posição indicada por HL**
3ØØ24	**DEC**	**HL**	**'2B'**	**; decrementa HL**
3ØØ25	**DEC**	**E**	**'1D'**	**; decrementa E que é utilizado como contador**
3ØØ26	**JP**	**NZ 3ØØ22**	**'C24675'**	
	ou JR	**NZ —6**	**'2ØFA'**	**; se E não for zero volte para colocar outro espaço em branco**
3ØØ29	**RET**		**'C9'**	

Tabela III

XX também em Linguagem de Máquina existem instruções para chamar DICIONALMENTE; como mostra a tabela V.

E como era de se esperar, podemos também fazer RET condicionalmente:

INSTRUÇÃO	CÓDIGO
RET Z	'C8'
RET NZ	'C0'
RET C	'D8'
RET NC	'D0'
RT PE	'E8'
RET PO	'E0'
RET M	'F8'
RET P	'F0'

Como funcionam estas instruções? Ao fazer um CALL o conteúdo do *contador de programa* que indica a próxima instrução (PC) é colocado no *topo da pilha* indicada pelo SP (seria como se existisse uma instrução PUSH PC) e o contador é então carregado com o endereço que está logo após o código de CALL que é o endereço onde começa a sub-rotina. Assim o computador passa a executar as instruções a partir desse endereço e, ao encontrar uma instrução de RET, o topo da pilha é recolocado no PC (seria como se fosse um POP PC) e portanto, o programa volta para a próxima instrução*após o CALL.*

Cuidados devem ser tomados sempre que, durante uma sub-rotina, mexermos no "topo da pilha", para evitar que o programa, ao encontrar um RET, volte para posição não desejável, isto porque o endereço de retorno da sub-rotina é colocado no topo da pilha pela instrução CALL.

Assim, se por ventura SP for alterado durante uma sub-rotina, deveremos GARANTIR que antes da instrução RET, seu valor seja igual ao seu valor no início da mesma. Note que esta estrutura permite que haja "sub-rotinas dentro de sub-rotinas". Muitas vezes ao chamar uma sub-rotina é necessário "salvar" os conteúdos de registros de alguns registros, o que pode ser feito usando PUSH. Lembre-se sempre então de fazer os POPs correspondentes (na ordem certa) *antes* de fazer RET. Para exemplificar, vamos fazer uma sub-rotina que "multiplica" dois números positivos menores que 256 que estão no *acumulador* e no *registro E* respectivamente; a sub-rotina deve colocar o resultado no par HL *sem alterar nenhum outro registro.* Naturalmente, como não temos uma instrução de multiplicação, faremos somas sucessivas. Usaremos um loop utilizando a instrução DJNZ que, como vimos anteriormente, funciona da seguinte maneira: decrementa o *registro B,* se o resultado for diferente de zero ela faz um *salto relativo* (+ 127 a — 128) para o endereço indicado, caso contrário prossegue o programa. Portanto, o registro B funciona como um *contador* e será alterado na sub-rotina, devendo então ser "salvo" (PUSH BC). Coloque a sub-rotina a partir do endereço 30100 e tente colocar você mesmo, os códigos em Linguagem de Máquina que mostramos na tabela VI.

Perceba que os PUSHs e POPs permitiram a condição de NÃO alterar nenhum outro registro. Veja que o número de PUSHs é igual ao número de POPs e com isto o valor de SP não é modificado, fazendo com que a sub-rotina volte para o endereço certo. Vamos tentar utilizá-la? Façamos um programa a partir da memória 30000 que multiplique dois números, por exemplo 15 e 24, usando a sub-rotina mostrada na tabela VII.

Verifique o resultado (que deve ser 360) e "rode" o programa usando XF no HEXAMEM. Aliás, quando não temos nenhum efeito especial de tela mas apenas cálculos, é conveniente executar os programas em FAST.

30022	LD	B,E,	'43'	; coloca o contador em B
30023	LD	(HL), 0	'3600'	
30025	DEC	HL	'2B'	
30026	DJNZ	−5	'10FB'	; decrementa o contador B e salta de volta se não for zero
30028	RET		'C9'	

Tabela IV

INSTRUÇÃO	CÓDIGO
CALL Z endereço	'CC' + + 2 bytes para endereço
CALL NZ endereço	'C4' + 2 bytes para endereço
CALL Z endereço	'DC' + 2 bytes para endereço
CALL NC endereço	'D4' + 2 bytes para endereço
CALL PE endereço	'EC' + 2 bytes para endereço
CALL PO endereço	'E4' + 2 bytes para endereço
CALL M endereço	'FC' + 2 bytes para endereço
....	....
....	....
CALL P endereço	'F4' + 2 bytes para endereço

Tabela V

30100	PUSH	BC	; "salva" B e C
30101	LD	B,E	; coloca um dos operandos no contador B
30102	PUSH	DE	; "salva" D e E
30103	LD	D,0	; coloca o outro operando no par DE
30105	LD	E,A	
30106	LD	HL,0	; coloca 0 no par HL
30109	ADD	HL,DE	; soma o segundo operando sucessivamente tantas vezes quantas indicar o primeiro operando B
30110	DJNZ	−3	
30112	POP	DE	; reestabelece os valores iniciais dos registros
30113	POP	BC	

Tabela VI

30000	LD	A,15	; carrega operando
30002	LD	E,24	
30004	CALL	30100	; chama sub-rotina de multiplicação
30007	LD	B,H	; programa a saída
30008	LD	C,L	
30009	RET		

Tabela VII

EXERCÍCIOS

1. Faça um programa que execute um LOOP utilizando a instrução INC. Por exemplo, mande escrever na tela os primeiros 20 caracteres do TK. Lembre-se que o endereço da primeira posição da tela está indicado nas memórias 16396 e 16397 devendo-se somar um.

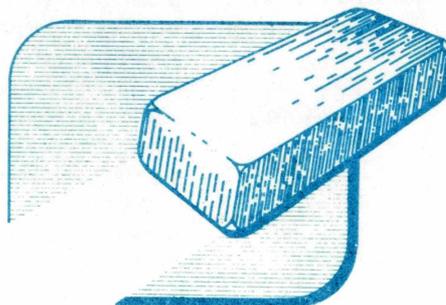
(Obs.: você deverá usar a instrução CP para verificar o fim do LOOP).

2. Faça uma sub-rotina que, por subtrações sucessivas, encontre o resultado INTEIRO da divisão de dois números positivos que estão no acumulador (dividendo) e registro E (divisor); o resultado deverá estar no registro D e nenhum outro registro deve ser alterado. A seguir, faça um programa que chame essa sub-rotina (veja o exemplo da multiplicação).

3. Faça uma sub-rotina em Linguagem de Máquina que some dois registros e "detecte" se houve ou não CARRY; em caso positivo, ela deve colocar o caractere C (código 4Ø ou '28') na tela na primeira coluna da segunda linha; caso contrário, ela deverá colocar o caractere N (código 51 ou '33') (Lembre-se de D-FILE).

4. Você saberia, usando quatro instruções colocar o ***valor*** de SP nos registros B e C? Feito isso, acrescente um RET e execute o programa para obter o valor de SP na tela.

5. Você saberia colocar o registro F no registro C? Feito isso, "zere" o registro B e acrescente um RET, executando o programa para obter o valor deste registro na tela.

○

Apagando linhas em bloco

Carlos Eduardo R. Salvato

No número 1 da revista, nós publicamos um programa ferramenta chamado RENUMERANDO que apresentava um inconveniente: logo após ser usado, você tinha que apagá-lo, linha por linha, o que tornava esta tarefa muito penosa.

Neste número nós estamos publicando um programa, quase todo em linguagem de máquina, que apaga um bloco de linhas por você especificado.

Digitação

Como todo programa em linguagem de máquina, você deverá ter um pouco de paciência ao digitá-lo. Apesar dele ser curto, todo cuidado é pouco.

Digite a linha 1 REM com 66 pontos e depois o programa da figura 1. Digite RUN e comece a entrar com os códigos da figura 2.

Quando você acabar, deverá aparecer na parte inferior do seu vídeo o código 9/7Ø.

Se isto não acontecer volte à fase anterior e redigite os códigos, até você conseguir acertar.

Grave o programa algumas vezes e digite RUN 8Ø.

Digite LIST e veja o que aconteceu: só ficou a parte que nos interessa sendo que, o resto já foi apagado, usando a própria rotina em linguagem de máquina.

Observação

A rotina em linguagem de máquina será preservada mesmo que você digite NEW, mas a parte em BASIC será perdida. Toda vez que você for usar o apagador de linhas só precisará carregar uma vez após ter ligado o computador, mas sempre que você digitar NEW terá que digitar o programa da figura 3.

→

```
   1 REM ..........................
..................................
...........
  10 FOR A=16514 TO 16579
  20 INPUT B
  30 SCROLL
  40 PRINT A;"...";B
  50 POKE A,B
  60 NEXT A
  70 STOP
  80 POKE 16389,127
  90 LET L=USR 16568
 100 POKE 32700,0
 110 POKE 32701,0
 120 POKE 32702,200
 130 POKE 32703,0
 140 LET L=USR 32512
9988 LET F=32700
9989 LET L=32702
9990 PRINT "APAGAR DE ";
9991 INPUT A
9992 POKE F,A-256*INT (A/256)
9993 POKE F+1,INT (A/256)
9994 PRINT A;" ATE" ";
9995 INPUT B
9996 POKE L,B-256*INT (B/256)
9997 POKE L+1,INT (B/256)
9998 PRINT B
9999 LET M=USR 32512
```

Figura 1

```
16514...42        16515...188
16516...127       16517...205
16518...216       16519...9
16520...229       16521...229
16522...42        16523...190
16524...127       16525...35
16526...205       16527...216
16528...9         16529...235
16530...42        16531...12
16532...64        16533...167
16534...237       16535...82
16536...46        16537...4
16538...237       16539...91
16540...12        16541...64
16542...42        16543...20
16544...64        16545...229
16546...167       16547...237
16548...82        16549...229
16550...193       16551...235
16552...241       16553...209
16554...245       16555...237
16556...176       16557...235
16558...209       16559...167
16560...237       16561...82
16562...229       16563...193
16564...225       16565...195
16566...173       16567...9
16568...33        16569...130
16570...64        16571...17
16572...0         16573...127
16574...1         16575...54
16576...0         16577...237
16578...176       16579...201
```

Figura 2

```
9988 LET F=32700
9989 LET L=32702
9990 PRINT "APAGAR DE ";
9991 INPUT A
9992 POKE F,A-256*INT (A/256)
9993 POKE F+1,INT (A/256)
9994 PRINT A;" ATE" ";
9995 INPUT B
9996 POKE L,B-256*INT (B/256)
9997 POKE L+1,INT (B/256)
9998 PRINT B
9999 LET M=USR 32512
```

Figura 3

MULTIESTOQUE

Nancy Mitie Ariga

TK-2000 foi lançado há pouco tempo e a Multisoft, paralelamente, desenvolveu vários programas para ele: Cadastro de clientes, Controle Bancário, Multiestoque, Multi-invader, Autoestrada e Corrida.

Alguns são aplicativos e outros jogos, porém, todos com versão em fita cassete para que o usuário não tenha a necessidade de comprar uma unidade de disco para carregá-los. Para aqueles que já possuem uma unidade de disco, a Multisoft promete que, em breve estes programas estarão disponíveis também em disquete.

Multiestoque

O programa Multiestoque foi um dos primeiros programas desenvolvidos para o TK 2000, já considerado um dos melhores para este computador.

É de fácil manuseio, voltado para as empresas que necessitam de um controle perfeito de seu estoque.

A fita é acompanhada por um manual que explica como utilizar o programa desde a sua carga até a emissão de relatório e gravação de um arquivo. O manual explica detalhadamente a função de todas as opções oferecidas pelo programa. No final, um exemplo prático é fornecido para que o usuário possa praticar a sua operação com um arquivo teste sem cometer erros, perdendo os dados.

Projetado para controlar até 850 itens por arquivo, este programa possibilita a emissão de relatórios das operações efetuadas no estoque (se a impressora estiver em ON-LINE), emissão da lista de preços, da posição do estoque e da lista de itens em posição crítica.

A organização dos registros no arquivo é feita em ordem alfabética baseando-se no código do item.

O programa ainda oferece a possibilidade de gravar em uma fita cassete o arquivo que está sendo manipulado pelo usuário e depois recuperá-lo através de uma leitura na fita cassete. Ou seja, o programa pode manipular um arquivo por vez, mas pode também, criar vários arquivos que serão gravados em fita cassete e, quando necessário, através de uma leitura, recuperar o arquivo desejado.

Carga do programa

Digite LOADT "ESTQ" ou LOADT para carregar o programa que está gravado no modo TK-2000. O volume do gravador deve estar entre 5 e 7. Verifique se os EARs do gravador e do TK-2000 estão conectados, pressione a tecla PLAY do gravador e a tecla RETURN do TK-2000.

Quando o programa começar a rodar aparecerá no final da tela:

ESTQ 2E ØØ WAIT

Depois de aproximadamente 70 segundos o programa está carregado e começa a rodar. A figura 1 mostra como a tela se apresenta. Digite a data da operação e pressione a tecla RETURN e a tela se apresentará conforme a figura 2.

O Arquivo

Como se pode observar no menu da figura 2, o programa possui várias funções que permitem a manipulação dos itens existentes no arquivo.

Um arquivo pode conter no máximo 850 itens que serão controlados pelo programa.

Um item é composto de vários campos cada um com uma função específica. São eles:

a) **código do item:** é um campo que contém o código referente a cada item afim de se manter uma organização no arquivo. É um campo alfanumérico com dez posições.

b) **descrição:** este campo contém

uma descrição de material em estoque. É um campo alfanumérico com 20 posições.

c) **atributo:** este campo é definido para conter informações adicionais como, por exemplo, código do fornecedor, unidade de medida, etc. É um campo alfanumérico com sete posições.

d) **preço de venda:** este campo contém o preço de venda do produto em estoque. É um campo numérico inteiro com oito posições.

e) **estoque mínimo:** este campo contém a quantidade mínima em estoque. É um campo numérico inteiro com quatro posições.

f) **saldo em estoque:** este campo contém o saldo atual em estoque. É um campo numérico inteiro com seis posições.

g) **valor em estoque:** este campo contém o valor total da mercadoria em estoque. É um campo numérico inteiro com oito posições.

h) **preço médio:** este campo é calculado pelo programa (valor em estoque ÷ saldo em estoque = preço médio). É um campo numérico decimal, com oito posições inteiras e duas decimais.

As figuras 3 a 10 ilustram algumas telas que podem ser obtidas com o uso deste programa.

A figura 3 ilustra uma tela de inclusão de um item preenchido. Todos os campos devem ser digitados pelo usuário.

A figura 4 mostra como a tela se apresenta ao escolher a opção 8 de alteração. Nesta opção, deve-se digitar o código do item a ser alterado, que os outros campos serão apresentados na tela se o código do item digitado existir no arquivo.

A figura 5 é um exemplo de como a tela se apresenta ao escolher a opção 9 de exclusão. Deve-se digitar o código do item a ser excluído e digitar S para confirmar a exclusão, ou N para que este item não seja excluído do arquivo.

A figura 6 mostra uma tela preenchida ao escolher a opção 1 para a *entrada* de mercadorias. Todos os campos devem ser preenchidos pelo usuário.

A figura 7 mostra como a tela se apresenta numa operação de *saída* (opção 2). Deve-se digitar o código do item, o número do documento e a quantidade. Os campos do preço e do total são preenchidos pelo programa. Essas duas opções permitem a movimentação de itens cadastrados no estoque.

```
MULTIESTOQUE(V1.2)
ITENS :    0              ARQUIVO :

                DATA :

(C)-1983 - MULTISOFT INFORMATICA LTDA.
```

Figura 1 — Inicialização de um arquivo

```
MULTIESTOQUE(V1.2)                      02/05/1984
ITENS :    0              ARQUIVO :
*MOVIMENTACAO        1.ENTRADAS
                     2.SAIDAS
*RELATORIOS          3.CONSULTA A ITEM
                     4.LISTA DE PRECOS
                     5.POSICAO DE ESTOQUE
                     6.SOMATORIAS
*MANUTENCAO          7.INCLUSAO
                     8.ALTERACAO
                     9.EXCLUSAO
*UTILITARIOS        10.GRAVACAO
                    11.LEITURA
                    12.REINICIALIZACAO
                    13.CONFIGURACAO
SELECIONE-->
(C)-1983 - MULTISOFT INFORMATICA LTDA.
```

Figura 2 — Menu

```
MULTIESTOQUE(V1.2)                 02/05/1984
ITENS :    0              ARQUIVO :
                INCLUSAO
    CODIGO ITEM       : AL000
    DESCRICAO         : TIJOLO COMUM
    PRECO DE VENDA    :  15000
    ESTOQUE MINIMO    :  100

    DADOS OK(S/N)?
(C)-1983 - MULTISOFT INFORMATICA LTDA.
```

Figura 3 — Inclusão

```
MULTIESTOQUE(V1.2)                 02/05/1984
ITENS :    0              ARQUIVO :
                ALTERACAO
    CODIGO ITEM       : AL000
    DESCRICAO         : TIJOLO COMUM
    PRECO DE VENDA    :  15000
    ESTOQUE MINIMO    :  100

    DADOS OK(S/N)?
(C)-1983 - MULTISOFT INFORMATICA LTDA.
```

Figura 4 — Alteração

```
MULTIESTOQUE(V1.2)                 02/05/1984
ITENS:    5               ARQUIVO :
                EXCLUSAO
    CODIGO ITEM       : AL000
    DESCRICAO         : TIJOLO COMUM
    PRECO DE VENDA    :  15000
    ESTOQUE MINIMO    :  100

    CONFIRMA (S/N)?
(C)-1983 - MULTISOFT INFORMATICA LTDA.
```

Figura 5 — Exclusão

```
MULTIESTOQUE(V1.2)                 02/05/1984
ITENS:    5               ARQUIVO :
                ENTRADAS
    CODIGO ITEM       : AL000
    DESCRICAO         : TIJOLO COMUM
    NUM. DOCTO.       : 1201/101
    QUANTIDADE        :  150
    VALOR             :  150000
    OPERACAO          :  3

    DADOS OK (S/N)?
(C)-1983 - MULTISOFT INFORMATICA LTDA.
```

Figura 6 — Entrada de item

```
MULTIESTOQUE(V1.2)                 02/05/1984
ITENS :    5              ARQUIVO :
                SAIDAS
    CODIGO ITEM  : AL000
    DESCRICAO    : TIJOLO COMUM
    NUM. DOCTO   : 1201/201
    QUANTIDADE   :       5
    VALOR        :     5000
    OPERACAO     : 2

    DADOS OK(S/N)?
(C)-1983 MULTISOFT INFORMATICA LTD.
```

Figura 7 — Saída de item

```
MULTIESTOQUE(V1.2)                 02/05/1984
ITENS :    5              ARQUIVO :
             CONSULTA A ITEM
    CODIGO ITEM      : AL000
    DESCRICAO        : TIJOLO COMUM
    ATRIBUTO         : MILHAR
    PRECO VENDA      :     15000
    ESTOQUE MIN.     :   100
    SALDO ESTOQ.     :     145
    VALOR ESTOQ.     :    145000
    PRECO MEDIO      :     1000,00
    < RETURN > MENU < ESPACO > PROXIMO
(C)-1983 MULTISOFT INFORMATICA LTD.
```

Figura 8 — Consulta a itens

```
MULTIESTOQUE(V1.2)                 02/05/1984
ITENS :    5              ARQUIVO :
             LISTA DE PRECOS

    CODIGO ITEM       : AL000

    DESCRICAO         : TIJOLO COMUM

    PRECO DE VENDA    : 15000

    < RETURN > MENU < ESPACO > PROXIMO
(C)-1983 MULTISOFT INFORMATICA LTD.
```

Figura 9 — Lista de preços

```
MULTIESTOQUE(V1.2)                 02/05/1984
ITENS :    5              ARQUIVO :
             POSICAO ESTOQUE
    CODIGO ITEM      : AL000
    DESCRICAO        : TIJOLO COMUM
    ATRIBUTO         : MILHAR
    PRECO MEDIO      :    1000,00
    ESTOQUE MIN.     :   100
    SALDO ESTOQ.     :     145
    VALOR ESTOQ.     :    145000
    < RETURN > MENU < ESPACO > PROXIMO
(C)-1983 MULTISOFT INFORMATICA LTD.
```

Figura 10 — Posição do estoque

```
M U L T I E S T O Q U E            RELATORIO DE INCLUSAO                PAG.: 1
(VERSAO 1.2)                             ARQUIVO:                       02/05/1984

      CODIGO             DESCRICAO              ATRIBUTO    PRECO VENDA     MIN.

      AL000          TIJOLO COMUM               MILHAR           15000       100
      AL005          TIJOLO BAIANO              MILHAR           35000       100
      AL010          TIJOLO APARENTE            MILHAR           50000       100
     .HD100          TORNEIRA PARA PIA          UNIDADE           6500       100
      FE010          PARAFUSO DE 1/10           UNIDADE             10      5000
```

Figura 11 — Relatório de Inclusão

```
M U L T I E S T O Q U E            RELATORIO DE ALTERACAO               PAG.: 1
(VERSAO 1.2)                             ARQUIVO:                       02/05/1984

      CODIGO             DESCRICAO              ATRIBUTO    PRECO VENDA     MIN.

      AL005          TIJOLO BAIANO              MILHAR           41000       100
      AL010          TIJOLO APARENTE            MILHAR           50000        50
```

Figura 12 — Relatório de Alteração

```
M U L T I E S T O Q U E            RELATORIO DE EXCLUSAO                PAG.: 1
(VERSAO 1.2)                             ARQUIVO:                       02/05/1984

      CODIGO             DESCRICAO              ATRIBUTO    PRECO VENDA     MIN.

      FE010          PARAFUSO DE 1/10           UNIDADE             10      5000
```

Figura 13 — Relatório de Exclusão

```
M U L T I E S T O Q U E            RELATORIO DE ENTRADAS                PAG.: 1
(VERSAO 1.2)                             ARQUIVO:                       02/05/1984

  OP.  NUM.DO DOC.      CODIGO         DESCRICAO              QTDE         VALOR

   1   1202/101         AL005        TIJOLO BAIANO             150        300000
   2   1203/101         HD100        TORNEIRA PARA PIA          50        250000
   3   1204/101         FE015        PARAFUSO DE 1/15         2000          2000
```

Figura 14 — Relatório de Entrada

```
M U L T I E S T O Q U E            RELATORIO DE SAIDAS                  PAG.: 1
(VERSAO 1.2)                             ARQUIVO:                       02/05/1984

  OP.  NUM.DO DOC.      CODIGO         DESCRICAO              QTDE         VALOR

   4   1202/201         FE015        PARAFUSO DE 1/16          100           100
```

Figura 15 — Relatório de Saída

```
M U L T I E S T O Q U E            POSICAO DE ESTOQUE                   PAG.: 1
(VERSAO 1.2)                             ARQUIVO:                       02/05/1984

     CODIGO          DESCRICAO          ATRIB.      PR.MEDIO    SALDO     VALOR  MIN.

  *  AL000       TIJOLO COMUM          MILHAR          0.00        0         0   100
     AL005       TIJOLO BAIANO         MILHAR       2000.00      150    300000   100
  *  AL010       TIJOLO APARENTE       MILHAR          0.00        0         0    50
  *  FE015       PARAFUSO DE 1/16      UNIDADE         1.00     1900      1900  5000
  *  HD100       TORNEIRA PARA PIA     UNIDADE      5000.00       50    250000   100
```

Figura 16 — Relatório de Posição do Estoque

```
M U L T I E S T O Q U E            LISTA DE PRECOS                      PAG.: 1
(VERSAO 1.2)                             ARQUIVO:                       02/05/1984

          CODIGO                        DESCRICAO                   PRECO VENDA

          AL000                     TIJOLO COMUM                         15000
          AL005                     TIJOLO BAIANO                        41000
          AL010                     TIJOLO APARENTE                      50000
          FE015                     PARAFUSO DE 1/16                        10
          HD100                     TORNEIRA PARA PIA                     6500
```

Figura 17 — Relatório de lista de preços

A figura 8 é um exemplo de como a tela se apresenta, escolhendo-se a opção 3 para a consulta a itens e digitando-se o código do item. Os outros campos são fornecidos pelo programa. Esta opção fornece os dados do item desejado.

A figura 9 é um exemplo de como a tela se apresenta ao escolher a opção 4 de lista de preços. É um relatório que pode listar os dados de todo o arquivo ou parte dele, podendo ser utilizado como um catálogo de preços para os vendedores. Se a impressora estiver OFF-LINE a lista de preços será apresentada na tela. Caso a impressora esteja em ON-LINE a lista de preços será apresentada na impressora como veremos adiante na figura 17.

A figura 10 é um exemplo de como a tela se apresenta ao escolher a opção 5 de posição do estoque.

As figuras 11 a 17 ilustram alguns relatórios que podem ser obtidos com o uso do programa Multiestoque se a impressora estiver em ON-LINE.

A figura 11 mostra como se apresenta o relatório de inclusão (opção 7).

A figura 12 mostra como se apresenta um relatório de alteração (opção 8) após preencher as fichas contendo os dados com as alterações.

A figura 13 mostra como se apresenta um relatório de exclusão (opção 9) após preencher as fichas dos itens a serem excluídos.

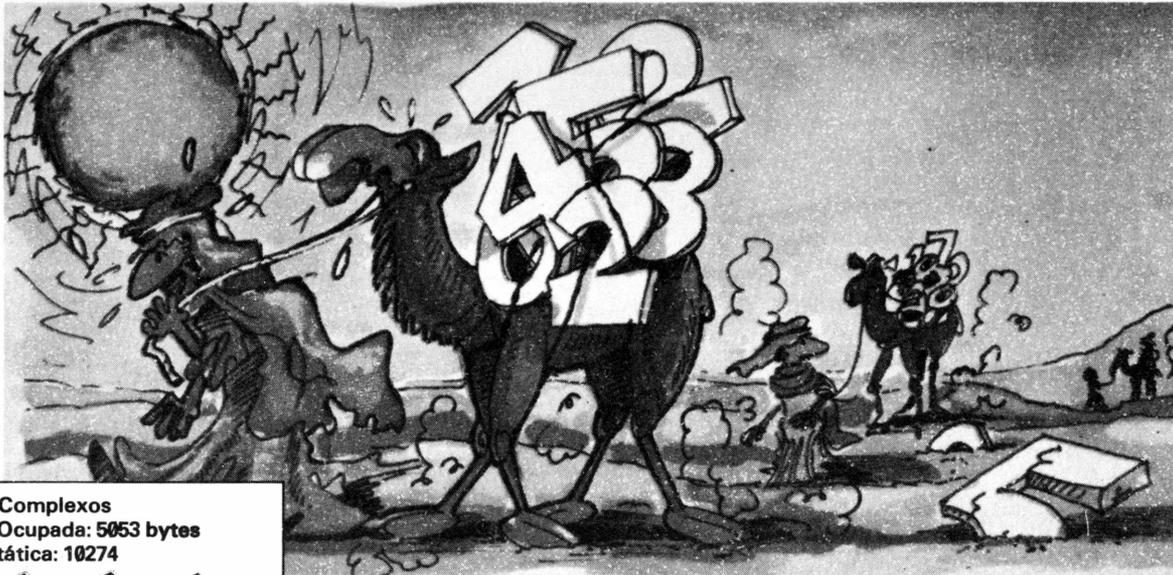
A figura 14 mostra como se apresenta um relatório de entrada (opção 1) ao preencher as fichas com os dados de entrada do produto.

A figura 15 mostra como se apresenta um relatório de saída de mercadorias (opção 2).

A figura 16 mostra como se apresenta um relatório de posição de estoque de todos os itens contidos no estoque (opção 5). Os itens que estão abaixo do estoque mínimo são destacados com um asterisco.

A figura 17 mostra como se apresenta um relatório de lista de preços de todos os itens contidos no arquivo (opção 4).

Como você pode notar, o Multiestoque é um programa complexo, porém de fácil manuseio, que visa um controle eficaz do estoque de uma empresa, oferecendo várias operações para a manipulação dos itens contidos no arquivo como por exemplo a alteração de um item ou a sua exclusão, movimentação, emissão de relatórios (de lista de preços, posição do estoque e dos itens abaixo do estoque), gravação de um arquivo em fita cassete e a leitura de um arquivo que foi gravado numa fita cassete.

NÚMEROS COMPLEXOS

Números Complexos
Memória Ocupada: 5053 bytes
Soma Sintática: 10274
Nível:
Computadores: TK-83 e compatíveis

José Gimenez

Este programa tem por finalidade resolver um sistema de ***n*** equações a ***n*** incógnitas, onde podemos trabalhar não apenas com números reais, mas também com números complexos na forma polar.

A principal utilização deste programa está na área eletro-eletrônica onde os números complexos são empregados largamente na representação de circuitos elétricos, que envolvam capacitâncias e indutâncias. Existem diversas maneiras para resolvermos tais circuitos, mas em todas elas poderemos construir um sistema matricial de equações em que as incógnitas são correntes de malhas ou tensões nodais. Desde simples circuitos eletrônicos até estudos de load-flow (fluxo de carga) em sistemas de distribuição de energia, este programa pode ser convenientemente utilizado.

Um número complexo é a representação de um número real e um número imaginário e pode ser escrito de duas maneiras:

— *forma retangular:* a + jb
onde *a* representa a parte real e *b* representa a parte imaginária
— *forma polar:*

$$\sqrt{a^2 + b^2}\ \underline{|\text{arctg } b/a}$$

$$\text{módulo}\ \underline{|\text{fase}}$$

Devido as características de resolução do programa, utilizaremos a forma polar e, para termos um número real puro, basta fazermos a parte imaginária igual a zero (b = 0) o que representa fase nula.

Um sistema matricial de equações pode ser escrito da maneira que mostramos na figura 1.

O método utilizado para a resolução do sistema é o método de eliminação de GAUSS ou de diagonização, que consta do seguinte:

1º passo — dividir a primeira linha por A(1,1), a segunda linha por A(2,1) e a n-ésima linha por A(I,1), de forma a obter o que mostramos na figura 2.

2º passo — subtrair a linha 2 da linha 1, a linha 3 da linha 1 e assim sucessivamente até subtrair a n-ésima linha da linha 1. Desta forma teremos o que mostramos na figura 3.

3º passo — dividir a linha 2 por A''(2,2), a linha 3 por A''(3,2) e assim até dividir a n-ésima linha por A''(I,2). Após isso, subtrair a linha 3, a linha 2, a linha 4 da linha 2 e sucessivamente até a n-ésima linha da linha 2, conforme o que aparece na figura 4.

Podemos notar que, seguindo esta seqüência de divisões e subtrações, vamos diagonalizar a matriz para obter o que mostramos na figura 5.

A primeira incógnita que tiraremos será V(I) e, por substituição, as demais.

Para diferenciarmos a introdução dados em módulo e fase, utilizamos o seguinte processo:

A(*1*,IJ), V(*1*,I) e P (*1*,I) onde o número *1* representa o *módulo,* e A(*2*,I,J), V(*2*,I) e P(*2*,I) onde o número *2* representa a *fase.*

O programa é dividido em sub-rotinas da seguinte maneira:

linhas 990 a 1600: explicações do programa, definição de variáveis e matrizes e introdução de dados.
linhas 1640 a 1690: transformação do ângulo de fase introduzido em graus para radianos.
linhas 1700 a 2060: faz a subtração entre dois números complexos transformando-os primeiramente para a forma retangular.
linhas 2070 a 2340: calcula os valores das incógnitas.
linhas 2360 a 2410: apresentação dos resultados.
linhas 2490 a 2580: faz a divisão entre dois números complexos.
linhas 2600 a 2950: bloco de correção de valores se forem introduzidas erroneamente.
linhas 2990 a 3030: bloco auxiliar de transformação polar/retangular.
linhas 3200 a 3210: gravação do programa.

$$\begin{bmatrix} A(1,1) & A(1,2) & \cdots & A(1,J) \\ A(2,1) & A(2,2) & \cdots & A(2,J) \\ A(3,1) & A(3,2) & \cdots & A(3,J) \\ \vdots & \vdots & & \vdots \\ A(I,1) & A(I,2) & \cdots & A(I,J) \end{bmatrix} \times \begin{bmatrix} V(1) \\ V(2) \\ V(3) \\ \vdots \\ V(I) \end{bmatrix} = \begin{bmatrix} P(1) \\ P(2) \\ P(3) \\ \vdots \\ P(I) \end{bmatrix}$$

onde V(1) a V(I) representam as incógnitas do sistema.

Figura 1

$$\begin{bmatrix} 1 & A'(1,2) & \cdots & A'(1,J) \\ 1 & A'(2,2) & \cdots & A'(2,J) \\ \vdots & \vdots & & \vdots \\ 1 & A'(I,2) & \cdots & A'(I,J) \end{bmatrix} \times \begin{bmatrix} V(1) \\ V(2) \\ \vdots \\ V(I) \end{bmatrix} = \begin{bmatrix} P(1) \\ P'(2) \\ \vdots \\ P'(I) \end{bmatrix}$$

Figura 2

$$\begin{bmatrix} 1 & A'(1,2) & \cdots & A'(1,J) \\ 0 & A''(2,2) & \cdots & A''(2,J) \\ \vdots & \vdots & & \vdots \\ 0 & A''(I,2) & \cdots & A''(I,J) \end{bmatrix} \begin{bmatrix} V(1) \\ V(2) \\ \vdots \\ V(I) \end{bmatrix} = \begin{bmatrix} P'(1) \\ P''(2) \\ \vdots \\ P''(I) \end{bmatrix}$$

Figura 3

$$\begin{bmatrix} 1 & A'(1,2) & \cdots & A'(I,J) \\ 0 & 1 & \cdots & A'''(2,J) \\ \vdots & \vdots & & \vdots \\ 0 & 0 & \cdots & A''''(I,J) \end{bmatrix} \times \begin{bmatrix} V(1) \\ V(2) \\ \vdots \\ V(I) \end{bmatrix} = \begin{bmatrix} P'(1) \\ P'''(2) \\ \vdots \\ P''''(I) \end{bmatrix}$$

Figura 4

$$\begin{bmatrix} 1 & A'(1,2) & \cdots & A'(1,J) \\ 0 & 1 & \cdots & A'''(2,J) \\ \vdots & \vdots & & \vdots \\ 0 & 0 & \cdots & 1 \end{bmatrix} \begin{bmatrix} V(1) \\ V(2) \\ \vdots \\ v(I) \end{bmatrix} = \begin{bmatrix} P'(1) \\ P'''(2) \\ \vdots \\ P'(I) \end{bmatrix}$$

Figura 5

Para ficar mais claro, vamos desenvolver um exemplo com o seguinte circuito elétrico, resolvendo-o por corrente de malha!

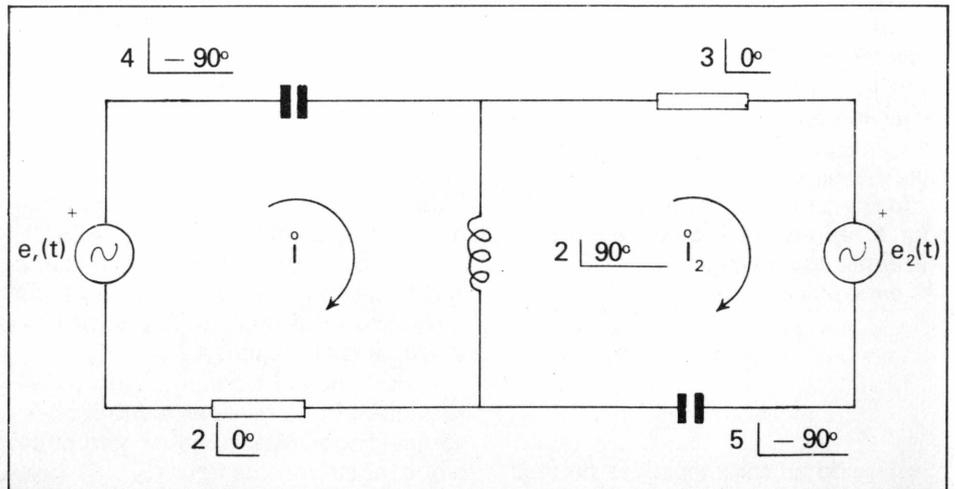



Figura 6

Dados:

$$\begin{cases} e_1(t) = 10\sqrt{2} \, . \, \cos 377t \, (V) \text{ ou } e_1 = 10\underline{|0^\circ} \, (V_{ef}) \\ e_2(t) = 5\sqrt{2} \, . \, \cos (377t + 90^\circ) \, (V) \text{ ou } e_2 = 5\underline{|90^\circ} \, (V_{ef}) \end{cases}$$

Montamos o sistema a duas **equações e a** duas **incógnitas:**

$$\begin{cases} (4\underline{|-90^\circ} + 2\underline{|0^\circ}) \, . \, I_1 - 2\underline{|90^\circ} \, . \, I_2 = 10\underline{|0^\circ} \\ -2\underline{|90^\circ} \, . \, I_1 + (2\underline{|90^\circ} + 5\underline{|-90^\circ} + 3\underline{|0^\circ}) \, I_2 = -5\underline{|90^\circ} \end{cases}$$

$$\begin{cases} 2{,}83\underline{|-45^\circ} \, . \, I_1 - 2\underline{|90^\circ} \, . \, I_2 = 10\underline{|0^\circ} \\ -2{,}0\underline{|90^\circ} \, I_1 + 4{,}24\underline{|-45^\circ} \, . \, I_2 = -5\underline{|90^\circ} \end{cases}$$

O programa é auto-explicativo e o leitor não terá dificuldades na entrada de dados.

Experimente introduzir os dados do exemplo:

```
A(1,1,1)  = 2.83
A(2,1,1)  = -45
A(1,1,2)  = -2
A(2,1,2)  = 90
P(1,1)  = 10
P(2,1)  = 0
A(1,2,1)  = -2
A(2,2,1)  = 90
A(1,2,2)  = 4.24
A(2,2,2)  = -45
P(1,2)  = -5
P(2,2)  = 90
```

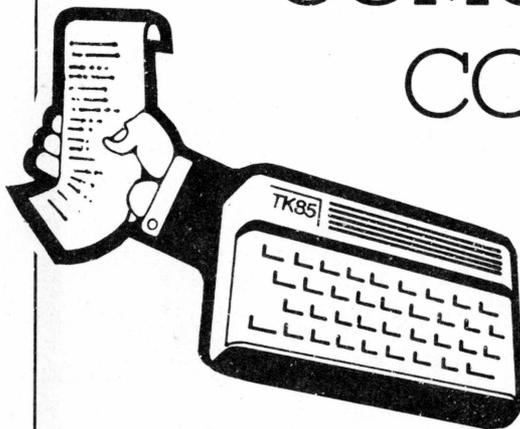
Que dará como resultado:

```
V(1,1)  = 3.9510175
V(2,1)  = 34.598289

V(1,2)  = 1.1181013
V(2,2)  = 206.59577
```

```
 990 SLOW
1000 REM ESTE PROGRAMA FOI ELABO
RADO POR JOSE ANTONIO GIMENES -
NOV/83
1010 PRINT "ESTE PROGRAMA RESOLV
E UM SISTEMADE EQUACOES COM NUME
ROS  COMPLE-XOS,NA SEGUINTE FORM
A MATRICIAL:"
1020 PRINT
1030 PRINT "■A(1,1)---- A(1,J)■■
V(1)■ ■P(1) ■"
1040 PRINT "■   /    ----     /  ■■
 /  ■ ■  /  ■"
1050 PRINT "■   /    ----     /  ■■
 /  ■=■  /  ■"
1060 PRINT "■   /    ----     /  ■■
 /  ■ ■  /  ■"
1070 PRINT "■A(I,1)---- A(I,J)■■
V(I)■ ■P(I) ■"
1080 PRINT
1090 PRINT "AS INCOGNITAS SAO: V
(1)---- V(I)"
1100 PRINT
1110 PRINT "OBRIGATORIAMENTE A(1
,1)<>0"
1120 PRINT
1130 PRINT "OS DADOS DE ENTRADA
SAO:"
1140 PRINT
1150 PRINT "A(1,I,J) E P(1,I) PA
RA MODULO, E"
1160 PRINT
1170 PRINT "A(2,I,J) E P(2,I) P/
FASE (GRAUS)"
1180 PRINT AT 21,0;"DIGITE O NO.
 DE INCOGNITAS (I)"
1190 LET K=1
1200 LET C=2
1210 INPUT N
1220 PRINT AT 21,29;"=";N
1230 PAUSE 120
1240 CLS
1250 FAST
1260 DIM A(1,N,N)
1270 DIM B(1,N,N)
1280 DIM A(2,N,N)
1290 DIM B(2,N,N)
1300 DIM M(N)
1310 DIM F(N)
1320 DIM P(1,N)
1330 DIM P(2,N)
1340 DIM V(1,N)
1350 DIM V(2,N)
1360 FOR I=1 TO N
1370 FOR J=1 TO N
1380 PRINT AT 21,0;"A(1,";I;",";
J;") = ";
1390 INPUT A(1,I,J)
1400 PRINT A(1,I,J)
1410 SCROLL
1420 PRINT "A(2,";I;",";J;") = "
;
1430 INPUT A(2,I,J)
1440 PRINT A(2,I,J)
1450 SCROLL
1460 NEXT J
1470 PRINT "P(1,";I;") = ";
1480 INPUT P(1,I)
1490 PRINT P(1,I)
1500 SCROLL
1510 PRINT "P(2,";I;") = ";
1520 INPUT P(2,I)
1530 PRINT P(2,I)
1540 SCROLL
1550 NEXT I
1560 SCROLL
1570 PRINT AT 21,0;"ALGUM VALOR
ERRADO (S/N)  "
1580 INPUT L$
1590 IF L$="S" THEN GOTO VAL "25
90"
1600 IF L$<>"N" THEN GOTO VAL "1
570"
1610 CLS
1620 LET S=1
1630 GOSUB VAL "2490"
1640 FOR I=1 TO N
```

```
1650 FOR J=1 TO N
1660 LET B(2,I,J)=B(2,I,J)*(PI/1
80)
1670 NEXT J
1680 LET P(2,I)=P(2,I)*(PI/180)
1690 NEXT I
1700 FOR I=C TO N
1710 FOR J=1 TO N
1720 IF B(1,I,S)=0 THEN GOTO VAL
 "3150"
1730 LET X=B(1,I,J)*COS B(2,I,J)
-B(1,K,J)*COS B(2,K,J)
1740 LET Y=B(1,I,J)*SIN B(2,I,J)
-B(1,K,J)*SIN B(2,K,J)
1750 LET A(1,I,J)=SQR (ABS X**2+
ABS Y**2)
1760 GOSUB VAL "2990"
1770 IF X=0 OR Y=0 THEN GOTO VAL
 "3040"
1780 LET A(2,I,J)=ATN (Y/X)
1790 IF X<0 THEN LET A(2,I,J)=A(
2,I,J)+PI
1800 NEXT J
1810 LET X=P(1,I)*COS P(2,I)-P(1
,K)*COS P(2,K)
1820 LET Y=P(1,I)*SIN P(2,I)-P(1
,K)*SIN P(2,K)
1830 LET P(1,I)=SQR (ABS X**2+AB
S Y**2)
1840 GOSUB VAL "2990"
1850 IF X=0 OR Y=0 THEN GOTO VAL
 "3060"
1860 LET P(2,I)=ATN (Y/X)
1870 IF X<0 THEN LET P(2,I)=P(2,
I)+PI
1880 NEXT I
1890 FOR I=1 TO K
1900 FOR J=1 TO N
1910 LET A(1,K,J)=B(1,K,J)
1920 LET A(2,K,J)=B(2,K,J)
1930 NEXT J
1940 NEXT I
1950 LET S=S+1
1960 GOSUB VAL "2490"
1970 FOR I=1 TO K
1980 FOR J=1 TO N
1990 LET B(1,K,J)=A(1,K,J)
2000 LET B(2,K,J)=A(2,K,J)
2010 NEXT J
2020 NEXT I
2030 LET C=C+1
2040 LET K=K+1
2050 IF S=N THEN GOTO VAL "2070"
2060 GOTO VAL "1700"
2070 LET Z=1
2080 FOR R=N TO 1 STEP -1
2090 FOR W=2 TO N
2100 IF B(1,W,Z)=0 THEN GOTO VAL
 "2960"
2110 IF R+W-1>N THEN PRINT "INVE
RTER AS EQUACOES DE ENTRADA DE M
ODO A TER A(1,1)<>0",,,"DIGITE ■
■■ NOVAMENTE"
2120 LET M(W)=B(1,R,R+W-1)*V(1,R
+W-1)
2130 LET F(W)=B(2,R,R+W-1)+V(2,R
+W-1)
2140 NEXT W
2150 LET X=0
2160 LET Y=0
2170 FOR W=2 TO N
2180 LET X=M(W)*COS F(W)+X
2190 LET Y=M(W)*SIN F(W)+Y
2200 NEXT W
2210 LET Q=SQR (ABS X**2+ABS Y**
2)
2220 GOSUB VAL "2990"
2230 IF X=0 OR Y=0 THEN GOTO VAL
 "2260"
2240 LET F=ATN (Y/X)
2250 IF X<0 THEN LET F=F+PI
2260 LET X=P(1,R)*COS P(2,R)-Q*C
OS F
2270 LET Y=P(1,R)*SIN P(2,R)-Q*S
IN F
2280 LET V(1,R)=SQR (ABS X**2+AB
S Y**2)
```

COMO COLABORAR COM MICROHOBBY

A Revista MICROHOBBY foi criada para servir de intercâmbio entre os leitores que participam do mágico mundo da computação.

A característica realmente inovadora do computador pessoal, está em transformar cada consumidor num criador. Aproveite sua criatividade e envie suas colaborações recebendo remuneração a título de DIREITO AUTORAL.

A maneira ideal de nos enviar o material a ser publicado obedece às seguintes normas:

1. Nunca esqueça de colocar o nome completo, telefone, endereço e número de sua assinatura em **todo** material enviada a nós, sejam listagens de impressora, fitas, envelope, carta ou qualquer outro material.

2. Envie a listagem de programa **datilografada** ou, melhor ainda, tirada na impressora do computador.

3. Coloque sempre uma linha REM com o nome do autor e o título do programa.

4. Envie uma fita com o programa gravado **algumas vezes** (se possível em gravadores diferentes).

5. Na fita, gravar **com microfone** (em viva voz), algumas instruções úteis:
nome completo e **endereço do remetente.**

6. Quando o programa for adaptado e/ou traduzido de outra revista, citar a fonte (autor original, data de publicação, nome da revista e todos os detalhes que houver referente à publicação).

7. Anexar ao material, uma carta autorizando a publicação por parte da revista e assumindo a responsabilidade pela autoria do material e/ou adaptações. Nesta carta, para agilizar a remuneração, podem constar os dados da conta corrente onde daremos o depósito correspondente aos direitos autorais.

8. O material **não utilizado não será devolvido,** ficando a critério da redação a decisão final sobre sua publicação.

9. O material deve ser enviado para:

MICROMEGA PUBLICAÇÕES E MATERIAL DIDÁTICO
SEÇÃO PROGRAMAS DO LEITOR
Cx. Postal 54096
CEP 01296

```
2290 GOSUB VAL "2990"
2300 IF X=0 OR Y=0 THEN GOTO VAL
 "3080"
2310 LET V(2,R)=ATN (Y/X)
2320 IF X<0 THEN LET V(2,R)=V(2,
R)+PI
2330 LET Z=Z+1
2340 NEXT R
2350 SLOW
2360 PRINT "          R E S U L T A D
OS    :",,,
2370 FOR I=1 TO N
2380 PRINT "V(1,";I;") = ";V(1,I
)
2390 PRINT "V(2,";I;") = ";V(2,I
)*180/PI
2400 PRINT
2410 NEXT I
2420 RAND USR 3292
2430 LET A(1,I,J)=0
2440 LET A(2,I,J)=0
2450 GOTO VAL "1790"
2460 LET P(1,I)=0
2470 LET P(2,I)=0
2480 GOTO VAL "1570"
2490 FOR I=S TO N
2500 FOR J=1 TO N
2510 IF A(1,I,S)=0 THEN GOTO VAL
 "3100"
2520 LET B(1,I,J)=A(1,I,J)/A(1,I
,S)
2530 LET B(2,I,J)=A(2,I,J)-A(2,I
,S)
2540 NEXT J
2550 LET P(1,I)=P(1,I)/A(1,I,S)
2560 LET P(2,I)=P(2,I)-A(2,I,S)
2570 NEXT I
2580 RETURN
2590 CLS
2600 PRINT "CORRECOES :"
2610 PRINT
2620 PRINT "A(X,L,C) = ?"
2630 PRINT
2640 PRINT "P(X,C) = ?"
2650 PRINT
2660 PRINT "DIGITE :[illegible] PARA MOD
ULO;"
2670 PRINT "        [illegible] PARA FAS
E;"
2680 PRINT "        [illegible] NO. DA LIN
HA E"
2690 PRINT "        [illegible] NO. DA COL
UNA."
2700 PRINT
2710 PRINT "DESEJA CORRIGIR A(X,
L,C) OU P(X,C) ? (A/P)"
2720 PRINT
2730 INPUT L$
2740 IF L$="P" THEN GOTO VAL "28
60"
2750 PRINT "A(";
2760 INPUT X
2770 PRINT X;",";
2780 INPUT I
2790 PRINT I;",";
2800 INPUT J
2810 PRINT J;") = ";
2820 INPUT V
2830 PRINT V
2840 LET A(X,I,J)=V
2850 GOTO VAL "2940"
2860 PRINT "P(";
2870 INPUT X
2880 PRINT X;",";
2890 INPUT I
2900 PRINT I;") = ";
2910 INPUT V
2920 PRINT V
2930 LET P(X,I)=V
2940 PRINT AT 21,0;"MAIS ALGUMA
CORRECAO (S/N) "
2950 GOTO VAL "1560"
2960 IF W<Z THEN GOTO VAL "2120"
2965 LET M(W)=0
2970 LET F(W)=0
2980 GOTO VAL "2140"
2990 IF X>=0 AND Y=0 THEN LET F=
0
3000 IF X<0 AND Y=0 THEN LET F=P
I
3010 IF X=0 AND Y>0 THEN LET F=P
I/2
3020 IF X=0 AND Y<0 THEN LET F=-
PI/2
3030 RETURN
3040 LET A(2,I,J)=F
3050 GOTO VAL "1800"
3060 LET P(2,I)=F
3070 GOTO VAL "1880"
3080 LET V(2,R)=F
3090 GOTO VAL "2330"
3100 FOR J=1 TO N
3110 LET B(1,I,J)=A(1,I,J)
3120 LET B(2,I,J)=A(2,I,J)
3130 NEXT J
3140 GOTO VAL "2570"
3150 FOR J=1 TO N
3160 LET A(1,I,J)=B(1,I,J)
3170 LET A(2,I,J)=B(2,I,J)
3180 NEXT J
3190 GOTO VAL "1880"
3200 SAVE "RESOLUCAO MATRICIAL C
MPLEX█"
3210 GOTO 990
```

SIMÉTRIKO

Renato da Silva Oliveira

SiméTriKo é um programa que, apesar de ocupar menos de 2 kBytes de RAM possui uma "capacidade criadora" quase que ilimitada. Ele gera aleatoriamente belíssimas figuras dentro de um quadrado de 21 linhas por 21 colunas no centro da tela. Essas figuras modificam-se constantemente e o usuário pode interferir e participar na confecção delas durante a execução do programa através de sete teclas: **Z, V, I, W, D, F e P.**

Digitando a tecla **Z**, uma cópia da tela é produzida na impressora. A tecla **V**, se pressionada, limpa totalmente o vídeo e a tecla **I** produz a sua **"inversão"** (a figura fica em negativo). Pressionando-se a tecla **W** os pontos passam a ser **"unplotados"** e dessa forma, se o vídeo não está **"invertido"**, a figura é lenta e aleatoriamente apagada. Pode-se, ainda, escolher o modo de operação do micro através das teclas **D** (para **SLOW**) e **F** (para **FAST**) e, finalmente, mantendo-se pressionada a tecla **P** a figura na tela permanece inalterada. Se você não consegue lembrar a função de cada tecla, rode o programa e use-as!

Tome muito cuidado ao digitar as linhas de Ø a 4. Para produzir a linha Ø, digite:

```
POKE 16510,0
```

Na digitação da linha 1, para introduzir **RETURN, POKE** e **LIST**, utilize-se da palavra-chave **THEN** e depois apague-a movendo o cursor para a esquerda e usando **RUBOUT**. (Se você está com dúvidas consulte a seção Desgrilando de Microhobby número 5).

Após o programa ser rodado, a linha 1 deve modificar-se um pouco (alguns de seus caracteres não aparecerão na listagem). A partir de então, se você quiser, poderá apagar as linhas 2, 3 e 4.

Adiante encontra-se a listagem do programa (figura 1) e uma seqüência da evolução de uma figura na tela (figura 2).

Para salvar o programa em fita, após digitá-lo completamente, comande **GOTO 37Ø**.

```
   0 REM "SIMETRIKO"
   1 REM E£RND7 -:5? RETURN ?C L
EN ?7$4 POKE ( LIST TAN
   2 POKE 16539,126
   3 POKE 16541,118
   4 POKE 16546,119
  10 LET C=21
  20 LET D=0
  30 LET E=D
  40 FOR G=E TO C*(1+3*RND)
  50 LET H=9*RND
  60 LET D=D+H*RND
  70 LET E=E+H*RND
  80 LET D=D AND D<=C
  90 LET E=E AND E<=C
 100 LET I=C+10
 110 IF INKEY$="W" THEN GOTO 210
 120 PLOT I+D,C+E
 130 PLOT I+D,C-E
 140 PLOT I-D,C+E
 150 PLOT I-D,C-E
 160 PLOT I+E,C+D
 170 PLOT I+E,C-D
 180 PLOT I-E,C+D
 190 PLOT I-E,C-D
 200 GOTO 290
 210 UNPLOT I+D,C+E
 220 UNPLOT I+D,C-E
 230 UNPLOT I-D,C+E
 240 UNPLOT I-D,C-E
 250 UNPLOT I+E,C+D
 260 UNPLOT I+E,C-D
 270 UNPLOT I-E,C+D
 280 UNPLOT I-E,C-D
 290 IF INKEY$="Z" THEN COPY
 300 IF INKEY$="V" THEN CLS
 310 IF INKEY$="F" THEN FAST
 320 IF INKEY$="D" THEN SLOW
 330 IF INKEY$="I" THEN LET L=US
R 16531
 340 IF INKEY$="P" THEN GOTO 290
 350 NEXT G
 360 GOTO 10
 370 SAVE "SIMETRIKO"
 380 PRINT AT 0,10;"""SIMETRIKO"
"""
 390 PRINT "D- SLOW ",,"F- FAST
",,"Z-COPIA",,"W-UNPLOT",,"P-INT
ERROMPE",,"V-LIMPA A TELA",,"I-I
NVERTE O VIDEO"
 400 PAUSE 240
 410 CLS
 420 GOTO 1
```

fig. 1

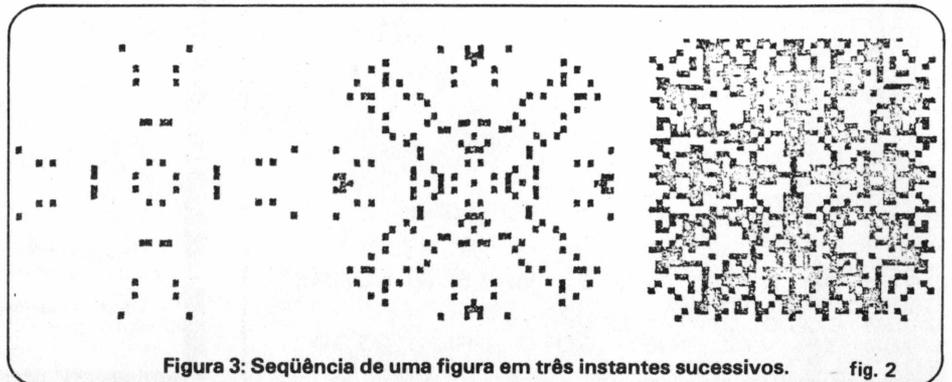



Figura 3: Seqüência de uma figura em três instantes sucessivos. fig. 2

Relação das lojas autorizadas a receber assinaturas

ALAGOAS

EXPOENTE COM. E REPRESENTAÇÕES LTDA. — Av. Siqueira Campos, 838 — Maceió.
JOALHERIA E BOUTIQUE GLORIA LTDA. — Pça. da Independência, 07 — Palmeira dos Índios — AL.

AMAZONAS

CENTRO DE ASSESSORIA EM PROC. DE DADOS LTDA. — Av. Costa e Silva, 680 — Manaus — AM.
PHONICA COM. IM. EXP. E REPRES. LTDA. — Rua Marechal Deodoro, 89 — Gr. 701/702 — 69.000 — Manaus — AM.

BAHIA

OFICCINA MINI E COMPUTADORES LTDA. — Shopping Center Itaigara — Lj. 40 — 1º Pav. — Salvador.
COMPUTER — FENA — COMP. E SIST. LTDA. — Av. Getúlio Vargas, 159 — Feira de Santana — BA.

CEARÁ

MICROCENTER COMPUTADORES LTDA. — Av. Santos Dumont, 2749 — Fortaleza.
SISCOMP SISTEMAS E COMPUTADORES LTDA. — Rua Tibúrcio Cavalcante, 298 — Fortaleza.

DISTRITO FEDERAL

ESCRITÓRIO DO FUTURO — MICROINFORMÁTICA, CONSULTORIA E TREINAMENTO — Conjunto Nacional — Brasília — 4º andar/4029 — Brasília — DF.

GOIÁS

CASA DO MICROCOMPUTADOR SIST. E PROC. DE DADOS LTDA. — Av. Anhanguera, 2574 — Goiânia.
NASA SHOP EQUIP. ELETRÔNICOS LTDA. — Rua 4, 1042 — Goiânia.

MINAS GERAIS

COMPUTRONIX VENDAS E SERVIÇOS LTDA. — Rua Sergipe, 1422 — Belo Horizonte.
DIDADOS INFORMÁTICA E ADM. LTDA. — Rua Minas Gerais, 655 — S/602 — Divinópolis.
DIVIDATA PROCESSAMENTO DE DADOS S/C LTDA. — Rua Rio de Janeiro, 1023 — C.P. 158 — Divinópolis — MG.
EXITUS INFORMÁTICA LTDA. — Rua Santo Antônio, 689 — Juiz de Fora — MG.
MAPSS — ENG. COM. E IND. LTDA. — Rua Getúlio Vargas, 186 — Teófilo Otoni.
MICROESPAÇO COM. E REPRESENTAÇÕES LTDA. — Av. Barão do Rio Branco, 1501 — Juiz de Fora.
MICRO POÇOS LTDA. — Rua Prefeito Chagas, 117 — Poços de Caldas — MG.
MICRON INFORMÁTICA LTDA. — Rua Benjamin Constant, 56 — S/804 — Viçosa.
MIKRO INFORMÁTICA LTDA. — Av. Afonso Pena, 952 — S/627 — Belo Horizonte — MG.
SIETEL SERV. INST. ELÉTRICAS E TEL. LTDA. — Rua Coronel Joaquim Neto, 32 — Santa Rita do Sapucaí.
MICRO E VÍDEO ELETRON. LTDA. — Av. Japão, 299 — Lj. 06 — Ipatinga — MG.
WORK ASSESSORIA DESENV. SISTEMA LTDA. — Rua Cruzeiro dos Peixotos, 499 — 9º andar — Sala 902 — Uberlândia — MG.
ZOG COMPUTADORES LTDA. — Rua Chagas Dória, 29 — Sala 303 — Lavras — MG.

MATO GROSSO DO SUL

D.R.L. ORG. EMPRESARIAL LTDA. — Av. Afonso Pena, 2081 — Lj. 9 — Campo Grande.

PARÁ

COMPUBEL — COMPUTADORES SIST. E SUPRIMENTOS LTDA. — Rua Quintino Bocaiuva, 1779 — Belém.
DISCOL DISTRIBUIÇÃO E COM. LTDA. — Rua 28 de Setembro, 746 — Belém.

PARANÁ

COMPUSHOP — Rua Emiliano Perneta, 509 — Caixa Postal 1522 — 80.000 — Curitiba — PR.
COMPUTE — COM. SUPRIMENTOS E EQUIP. ELETRÔNICOS LTDA. — Rua Cruz Machado, 474 C.P. 1427 — Curitiba — PR.
COMPUSTORE — Rua Emiliano Perneta, 509 — Curitiba.
GRUPO D.G.B. CONSULTORIA ADM. EMPRESARIAL S/C LTDA. — Rua Dr. Murici, 706 — Ala B — 1º andar — Curitiba.
MADISON S/A IMPORTAÇÃO E COMÉRCIO — Rua Mal. Deodoro, 311 — Curitiba — PR.
MORGEN — COM. DE COMPUTADORES — Rua Mal. Deodoro, 51 — Galeria Ritz, 14º and. — S/1405-A — Curitiba.
SHOP COMPUTER CEDM LTDA. — Av. São Paulo, 718 — Londrina.

PERNAMBUCO

ELETROSOM LTDA. — Rua da Concórdia, 287 — Recife.
ELÓGICA MICRO SISTEMAS LTDA. — Rua da Hora, 88 — Recife.
NOVA ERA MICROINFORMÁTICA — Rua Moisés Correa e Silva, 60 — Recife.
SOUZA'S COMPUTER CENTER LTDA. — Rua Mª Carolina, 205 — Loja 05 — Boa Viagem — Recife — PE.
TELEVÍDEO LTDA. — Rua Marques do Herval, 157 — Recife.

RIO DE JANEIRO

BEL-BAZAR ELETRÔNICO LTDA. — Av. Almirante Barroso, 81 — Lj. C — Rio de Janeiro.
BRASIL TRADE CENTER COM. E PARTICIPAÇÕES S/A — Av. Epitácio Pessoa, 280 — Rio de Janeiro.
CESPRO — CURSOS DE ESPECIALIZAÇÃO PROFISSIONAL LTDA. — Rua República Árabe da Síria, 15 — S/207 — Rio de Janeiro.
COMPUTER CENTER MICROCOMPUTADORES MÁQ. E SISTEMAS LTDA. — Rua Lopes Trovão, 134 — Slj. 247 — Center V — Niterói.
CONSISTEM — CONSULTORIA E TEC. EM MICROINFORMÁTICA LTDA. — Rua do Catete, 311 — Sala 318 — Rio de Janeiro — RJ.
DATAMICRO INFORMÁTICA LTDA. — Rua Visconde de Pirajá, 547 — Slj. 211 — Ipanema — Rio de Janeiro.
DATASERV SERVIÇOS DE COMPUTADORES LTDA. — Rua 151, 146 — Volta Redonda — RJ.
ELDORADO COMPUTADORES E SIST. LTDA. — Rua Visc. de Pirajá, 351 — Lj. 213/214 — Rio de Janeiro.
FOTO ÓTICA REPROPOLIS LTDA. — Rua do Imperador, 715 — Petrópolis.
KRISTIAN TELECOMUNICAÇÕES LTDA. — Rua da Lapa, 120/505 — Rio de Janeiro.
MASER — MÁQ. E SERV. DE PROCESSAMENTO DE DADOS — Estr. da Cacuia, 231 — Ilha do Governador — RJ.
MICROBYTE SISTEMA E EQUIPAMENTOS COMPUTACIONAIS LTDA. — Rua Buenos Aires, 41 — 3º and. — Rio de Janeiro.
MICRO CENTER INFORMÁTICA LTDA. — Rua Conde do Bonfim, 229 — Lj. 310/2 — Rio de Janeiro.
MICRO HOUSE REPR. LTDA. — Rua Visconde de Pirajá, 547 — S/307 — Rio de Janeiro.
MICROTRONICS COM. SIST. DIGITAIS LTDA. — Rua 16 de Março, 8A S/L 3 — Petrópolis — RJ.
POLIGAME VÍDEO CLUB LTDA. — Estrada do Galeão, 2979 S/305 — Rio de Janeiro — RJ.
PROSERV PROC. DE DADOS CURSOS E REP. LTDA. — Lgo. 9 de Abril, 27 — S/628 — Volta Redonda.
SINCLAIR PLACE DO BRASIL COMÉRCIO DE MICROCOMP. LTDA. — Rua Dias da Cruz, 215 — Sala 804 — Rio de Janeiro.
TELEMÁTICA COM. E IND. LTDA. — Rua Figueiredo de Magalhães, 286 — S/611 — Rio de Janeiro.

RIO GRANDE DO NORTE

INTERMIDIA COMPUTADORES E INFORMÁTICA LTDA. — Av. Nascimento de Castro, 1913 — Lagoa Nova Natal — RN.

RIO GRANDE DO SUL

INFORMÁTICA DINÂMICA LTDA. Rua Minas Gerais, 56 — Caixa Postal 289 — 98.900 — Santa Rosa — RS.
MAURITINO PIRES SILVEIRA — rua Manduca Rodrigues, 924 — Sant'Ana do Livramento.
METALDATA ENG. E PROC. LTDA. — Rua Álvaro Chaves, 154 — Cj. 302 — Porto Alegre.
MICROCENTER COMPUTADORES LTDA. — Rua 1º de Março, 113 — Ed. Integral, Cj. 202 — São Leopoldo — RS.
MICROMEGA COMP. E SISTEMAS LTDA. — Rua Julio de Castilhos, 441 — 1º and. — Novo Hanburgo — RS.
SIERRA REPRESENTAÇÕES — Av. Farrapós, 2287 — Porto Alegre.
SISTEMÁTIKA COMPUTADORES E SISTEMAS — Rua Andrade Neves, 248 — Pelotas.
VON EYE & CIA. LTDA. — Praça da República, 101 — 98.700 — Ijuí — RS.

SANTA CATARINA

COMPUTERVILLE MICROCOMPUTADORES LTDA. — Rua Tijucas, 375 — Joinvile.
CETIL MICROCOMPUTADORES S/A — Rua Itajaí, 1335 — Blumenau — SC.
ENTEC REPRESENTAÇÕES LTDA. — Rua Lauro Muller, 700 — Itajaí.
MICRODADOS COMP. SERV. LTDA. — Rua Anita Garibaldi, 8 — Slj. 1 e 2 — Florianópolis.
MICRO HOME MICROCOMPUTADORES E SISTEMAS LTDA. — Rua Pedro Soares, 9 — Florianópolis — SC.
SOME — SOC. MERCANTIL E INDL. LTDA. — Rua 15 de Novembro, 1139 — Blumenau.

SÃO PAULO

A.D. DATA COM. SERV. DE INFORMÁTICA LTDA. — Rua João Ramalho, 818 — São Paulo.
ACACIA COM. EXPORT. E IMPORTAÇÃO LTDA. — Av. Paulista, 2073 — Cj. 216/7 — São Paulo.
AGÊNCIA AVANT-GARDE — Av. Brig. Faria Lima, 1237 — Lj. 07 — São Paulo.
ALLCOLOR COM. E REPRESENTAÇÕES LTDA. — Rua Carlos Porto, 85 — Jacareí.
BENNY FEIRA PERMANENTE DE MICROCOMPUTADORES LTDA. — Rua Domingos de Moraes, 407 — São Paulo — SP.
CENADIN — CENTRO NAC. DESENV. DA INFORMÁTICA — Rua José Maria Lisboa, 580 — São Paulo.
CHIP SHOP COMPUTADORES LTDA. — Rua Ofélia, 248 — São Paulo.
COMPUTEC LTDA. — Rua Mairinque, 66 — São Paulo.
COMPUTER HOUSE — JOÃO CANDIDO COLLADO — Av. Andrade Neves, 1254 — Campinas.
COPEC COMPUTADORES, PROGRAMAS E COM. S/A — Rua Prof. Carlos dew Carvalho, 164 — 6º and. — São Paulo.
DATA SOLUTION LTDA. — Av. Eusébio Matoso, 654 — São Paulo.
ENSICOM — ENG. DE SISTEMAS E COMP. — Rua Marques do Herval, 409 — 1º andar — S/15 — Taubaté.
EXATRON INFORMÁTICA E ELETRÔNICA LTDA. — Al. dos Arapanés, 841 — São Paulo.
FRANCINI CENEVINA CIA. LTDA. — Rua 15 de Novembro, 1289 — Sala 06 — 13.560 — São Carlos.
GUAÇUMAQ — MÁQ. E EQ. P/ ESCRITÓRIOS LTDA. — Rua Antonio Gonçalves Teixeira, 97 — Mogi Guaçu.
GUARANI PRESENTES — Av. Senador Vergueiro, 4964 — 1º andar — S/6 — São Bernardo do Campo.
HECTOR A. FERNANDEZ — MIRAGE CINE FOTO — Rua Gal. Câmara, 648 — Santa Bárbara D'Oeste.
INFOR-MATIC INFORMÁTICA E AUTOMAÇÃO LTDA. — Av. Açocê, 309 — São Paulo.
LIDADOS SERVIÇOS E COM. DE COMPUTADORES LTDA. — Rua 7 de Setembro, 876 — Limeira.
LIVRARIA POLIEDRO — Rua Aurora, 704 — São Paulo.
LOG COMPUTADORES LTDA. — Pça. Cândido Dias Castejon, 34 — Slj. — São José dos Campos.
MEMOCARDS — MATERIAIS DIDÁTICOS LTDA. — Rua Amador Bueno, 855 — Ribeirão Preto.
MICRODATA IMPLANTAÇÃO FÍSICA SIST. LTDA. — Rua Montreal, 16 — São Paulo.
MICROCOMP COMPUTADORES LTDA. — Av. Pedroso de Moraes, 1234 — São Paulo.
MICRO PROCESS COMPUTADORES LTDA. — Al. Lorena, 1310 — Lj. 3 — São Paulo.
NÚCLEO DE ORIENTAÇÃO DE ESTUDOS — Av. Brig. Faria Lima, 1451 — S/31 — São Paulo.
NADAIS EQUIP. DE SOM LTDA. — Rua Amador Bueno, 213 — Santos.
MULTISOFT INFORMÁTICA LTDA. — Av. Angélica, 2318 — 13º andar — São Paulo.
PRO-ELETRÔNICA COML. LTDA. — Rua Santa Ifigênia, 568 — São Paulo.
RC MICROCOMPUTADORES LTDA. — Av. Estados Unidos, 983 — Piracicaba.
RITZ CINE FOTO LTDA. — Rua Frei Caneca, 7 — Santos.
SCHOCK ELETRÔNICA LTDA. — Rua Pe Luiz, 278 — Sorocaba.
PRO COMPUTADORES LTDA. — Av. Carlos Gomes, 396 — Marília.
TELEDALTO ELETRÔNICA E TELECOM. LTDA. — Rua 13 de Maio, 271 — S/101 — Catanduva.
TWIQUI COMPUTADORES E ACESSÓRIOS LTDA. — Av. Paulo Faccini, 72-A — S/6 — Guarulhos.

Os melhores programas para você.
PARA TK 2000 COLOR
MICROSOFT
MICROSOFT
Garantia integral
MICROSOFT
MICROSOFT
MICROSOFT
MICROSOFT
A Microsoft tem 120 programas em fitas e disquetes à sua disposição. São sistemas aplicativos para acompanhar e agilizar os negócios de sua empresa. E também jogos eletrônicos para você e sua família se divertirem muito. Todos especiais para TK-83, TK-85, TK-2000, Apple II e compatíveis. E todos com a mesma qualidade dos 100.000 programas já vendidos em todo o Brasil.
Procure o revendedor Microsoft mais próximo (se não encontrar os programas Microsoft escreva para a Caixa Postal 54221 - CEP 01000- S. Paulo-SP). Você encontrará os melhores programas da sua vida.
MICROSOFT®
Sempre o melhor programa.